

NASA Technical Memorandum 100640

EAC: A PROGRAM FOR THE ERROR ANALYSIS OF STAGS RESULTS FOR PLATES

(NASA-TM-100640) EAC: A PROGRAM FOR THE
ERROR ANALYSIS OF STAGS RESULTS FOR PLATES
(NASA) 74 p CSCI 20K

N89-17898

63/39 Unclass
0189709

Rajaram Sistla, Gaylen A. Thurston and Nancy Jane C. Bains

January 1989



National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665-5225

CONTENTS

Section	Page
INTRODUCTION	1
1. STAGS DATA.	2
Plate Geometry	2
Stiffness Properties	2
Boundary Conditions	3
2. Postprocessing STAGS Data	4
Input	5
FEM Input	5
Input to NPOSTPX.	5
Additional Input File	8
Input to Plotting Postprocessor EACPL	11
Procedures	13
Working On The Cyber.	13
Working On The MicroVAX	23
Executing STAGS On The MicroVAX	23
Sample STAGS Run On The MicroVAX.	24
Sample POSTP Run On The MicroVAX.	25
Executing EAC On The MicroVAX	26
Sample EAC Run On the MicroVAX.	27
Interactive Usage Of EAC.	28
Modifying EAC For Large Grids	28
3. Theory.	30
Constitutive Relations	30
Equilibrium Equations.	31
Displacement Equations and	
Boundary Conditions.	31
Continuous Solution For W.	32
Boundary Conditions for W.	34
Continuous Solutions For U and V	36
Output And Error Analysis.	38
4. TP User's Instructions.	41
Working on the Cyber	41
Working on the MicroVAX.	46
APPENDIX A : NOMENCLATURE.	47
APPENDIX B : EXAMPLE	48
References	56
Tables	57
Figures.	58

INTRODUCTION

Estimating the accuracy of stresses and deflections computed by the Finite Element Method (FEM) is a common problem that arises with each new application of the method. A computer code is now available for estimating the error in results from the STAGS finite element code (Ref. 1) for a shell unit consisting of a rectangular orthotropic plate. This memorandum describes the connection between the input data for the STAGS "shell units" and the input data necessary to run the error analysis code (called EAC). The STAGS code returns a set of nodal displacements and a discrete set of stress resultants. The continuous solution is defined by a set of generalized coordinates computed in EAC. The theory and the assumptions that determine the continuous solution are also outlined in this memorandum.

A structure, such as a stiffened panel, is modeled in STAGS as a number of shell units and/or finite elements. The current version of EAC performs error analysis on results for a rectangular plate unit from STAGS. The theory in the error analysis requires input data for the plate geometry, the material properties, and a set of boundary conditions. These input data come from the STAGS code. The specific link between the input and output data from STAGS and the input data for EAC is POSTP, the postprocessor program in the STAGS processors. A modified version of POSTP is described in Section 2 of this memorandum, following a brief review of the general procedure for executing EAC as part of the STAGS processors in Section 1.

Section 3 presents the theory for an orthotropic, rectangular plate with initial geometric imperfections. The analysis for the continuous solution is outlined, and the notation for the output data from EAC is defined. The error analysis is for a postbuckled plate. The postbuckled finite element solution in STAGS can be computed using the new TP option in STAGS (Ref. 2). Input for running the TP option in STAGS is summarized in Section 4. Finally, input and output for a sample problem are contained in Appendix B.

SECTION 1: STAGS DATA

Most of the input data for EAC is STAGS input or output data. The STAGS processors consist of STAGS1, STAGS2, POSTP, and STAPL. The input data for the plate geometry are part of the STAGS1 input. The material properties and lay-up for an orthotropic plate are also part of the usual STAGS1 input data. The error analysis in EAC uses STAGS2 output for edge displacements and rotations as boundary conditions. A complete description of STAGS input is found in the STAGS Users Manual, Ref. (1). Only input directly connected to the error analysis is reviewed here, the input data for the finite element results in STAGS are unchanged from the current version of STAGS.

1.1 Plate Geometry

In STAGS a structure, such as a stiffened panel, is modeled as a set of "shell units" and/or a set of finite elements. An execution of EAC is concerned with a single shell unit, a rectangular plate. The geometry of the plate is defined as STAGS1 input. The input follows the format starting with ISHELL=2 on the M-1 record. The length and width of the plate are deduced from data on the M-2A record. The logic in EAC assumes that a standard rectangular grid of nodal points is defined for the plate in STAGS. The number of rows in the rectangular grid for shell unit I is NRWS(I) on record F-1, and the corresponding number of columns is NCLS(I). There are several options in STAGS for the definition of initial imperfections over a flat plate. Any of these options can be used in the FEM analysis, however, the logic in EAC assumes a continuous function for the imperfections. Therefore, input for the definition of imperfections is required both in the STAGS1 input data and for EAC (see instructions for the EAC-6 record in Section 2).

1.2 Stiffness Properties

The theory in Section 3 of this memorandum defines stiffness coefficients A_{11} , A_{12} , A_{22} , A_{66} , D_{11} , D_{12} , D_{22} , and D_{66} for an orthotropic plate. These stiffnesses are computed by STAGS1 from the input in the K records and IWALL in the M-5 record.

1.3 Boundary Conditions

The boundary conditions for the error analysis in EAC for a plate shell unit are displacements and rotations on the edges of the plate computed in STAGS2. The modification of the postprocessor POSTP to manage these data in EAC is discussed in Section 2. The STAGS1 input data for boundary conditions and loads for the structure are unchanged; the specific plate unit considered in EAC has the usual input for boundary conditions, loads, and connection of boundary lines.

After completing the input data for STAGS1 for the shell structure containing the plate unit, STAGS1 and STAGS2 are executed to compute a finite element solution for the shell structure. The error analysis in EAC is then executed as a postprocessor of STAGS2 data. The data that must be extracted for the plate unit in EAC are the geometry, the stiffnesses, and a set of boundary conditions. The details of the postprocessing to obtain input for EAC are contained in the next section.

SECTION 2: POSTPROCESSING STAGS DATA

The error analysis code EAC requires a few new and some modified input data files and procedures. Detailed instructions are provided in this section to help the user prepare the various input files. Example procedure files are also provided for the execution of the STAGS postprocessors on the Cyber and MicroVAX machines at NASA Langley Research Center [LARC].

As mentioned earlier in Section 1, EAC uses as input, selected data from the input to and output from a STAGSC-1 analysis. This input file, called TAPE43, is prepared by using NPOSTPX, a modified version of the STAGS postprocessor POSTP. POSTP is used following a previous STAGSC-1 execution to enable the user to print quantities not selected for output in the original execution. NPOSTPX is POSTP modified to output selected data to a separate file, TAPE43. NPOSTPX requires a separate input file to specify the particular plate unit in the structure being analyzed, and the load step at which the data are to be written to TAPE43.

In its present version, EAC can handle only one plate unit, at one load step of the STAGS execution, using the non-corotation option. The FEM solution from STAGS is assumed to be a valid zeroth approximation for starting the iterative solution of the shell equations. This solution forms the first step of execution of EAC, during the first phase of which the resulting general solution in the form of Fourier coefficients and constants of integration (generalized coordinates) are written to a local file called TAPE45. During the second phase, these data are read from TAPE45 and are used to generate results at specified points on the plate, or over a user defined grid. There exists an option in EAC which permits the user to reuse data on TAPE45 saved from a previous EAC execution to generate output data over a different output grid or at selected points on the plate unit. Section 3 contains a summary of the numerical analysis in EAC that results in the generalized coordinates for the continuous solution. Results are written to two local files - TAPE6 and TAPE46.

TAPE46 is the input data file for EACPL, a contour plotting post-processor for EAC. TAPE46 contains all the functional data for plotting two

dimensional contour plots. Currently EACPL uses the USCONTOUR plotting package on the Cyber and DI3000 contour package on the MicroVAX. EACPL requires an additional input file called TAPE47 to provide plotting details such as the contour limits, intervals, etc. Examples for the MicroVAX are also provided in subsections titled WORKING ON THE MicroVAX.

2.1 INPUT

2.1.1 FEM INPUT

EAC requires as input, a local file called TAPE43. This is a file of selected discrete results from STAGS to be analyzed. TAPE43 is obtained by executing a modified version of POSTP after a previous STAGSC-1 execution. The input required for this new version of POSTP (called NPOSTPX) is described in the next sub-section. A sample procedure file for the generation of a TAPE43 is included in Section 2. STAGS results selected for TAPE43 are displacements, rotations, stress and moment resultants, strains, changes in curvatures. As with POSTP, the NPOSTPX execution requires a Solution Data file (SOD file) generated by a previous STAGS execution. On the Cyber, this file is called TAPE22 and must be available as a local file named TAPE20. A review of the POSTP input data cards as applicable to NPOSTPX is given next.

Note: The equivalent of NPOSTPX on the VAX is POSTP.EXE

2.1.2 INPUT TO NPOSTPX

TAPE43, the input data file for EAC, is created by executing NPOSTPX. NPOSTPX is a modified version of POSTP and was designed to create a TAPE43 as well as to select data created and saved by a previous STAGS execution for printing or plotting. It uses as input, a subset of the POSTP input. Just as with POSTP, NPOSTPX can be repeatedly executed with the same STAGS SOD data, each time with a different selection of output parameters. This would be useful in the preparation of TAPE43s for a series of EAC executions for different plate units in the same structure. POSTP users are directed to the STAGS Users Manual, Ref. 1, for input instructions. All input records to NPOSTPX are in STAGS free field format and reviewed below.

- Note 1: EAC can only be used for one plate unit and one load step at a time.
- Note 2: The co-rotation option in STAGS is incompatible with EAC, because edge rotations from the co-rotational option are not defined with respect to the local fixed coordinate system.
- Note 3: When used solely for the creation of a TAPE43, the user needs to specify only the first four records PP-1 to PP-4, with the first four variables in the PP-4 record selected as unity for the plate unit under consideration, keeping all others zero.

PP-1 Case Title Record

<u>Variable</u>	<u>Description</u>
COMENT	Arbitrary 80 character alphanumeric field to be printed as case title.

PP-2 Number of Load Steps

<u>Variable</u>	<u>Description</u>
NSTEPS	Number of load steps for which postprocessing is desired. Currently, NSTEPS-1.

PP-3 Load Step Sequence

<u>Variable</u>	<u>Description</u>
ISTEP	Integer load step in STAGS to be used for error analysis.

The next three records must be repeated for each shell unit in the structure. However, data to be printed must be selected only for the shell unit for which the TAPE43 is required.

PP-4 Output Control Parameters

<u>Variable</u>	<u>Description</u>
PRD	0 - Do not print displacements 1 - Print displacements for the load step.
IPRR	0 - Do not print stress resultants 1 - Print stress resultants

IPRE	0 - Do not print strains 1 - Print strains and changes of curvature
IPRS	0 - Do not print stresses 1 - Print stresses
IPRP	0 - for NPOSTPX.
IPRF	0 - Do not print point or nodal forces 1 - Print nodal forces
NSELD	The number of records defining selective displacements
NSELS	The number of records defining selected stresses.
IPRSDP	0 - Print selected displacements at the load step
IPRSTR	0 - Print selected stresses at the load step
ISL	0 - Stresses, strains and resultants are computed at element centroids. 1 - Stresses, strains and resultants are computed at element integration points.
ISS	0 - This entry is zero always.
ISD	0 - Print stresses or strain components in the directions of the shell. 1 - Print stress components in both the shell and the material coordinate directions.

NSELD > 0, go to PP-5

NSELD = 0, NSELS > 0, go to PP-6.

PP-5 Output Control Record 2

This record defines a number of nodes at which displacements are to be printed at each load or time step. If IROWD is set equal to zero the displacements are printed for each row, that is, the entire column. If ICOLD is set equal to zero the displacements are printed for each column, that is, the entire row. The record is repeated NSELD times (PP-4).

<u>Variable</u>	<u>Description</u>
IROWD	Row number at which displacements are to be printed.
ICOLD	Column number at which displacements are to be printed.

NSELS > 0, go to PP-6.

NSELS = 0, follow instructions at end of PP-6.

PP-6 Output Control Record 3

This record defines a number of locations at which stresses will be printed at each load or time step. Stress output will be consistent with the description for ISD on the PP-4 record or in a user-written subroutine. The record is repeated NSELS times (R-1 record of STAGS input).

<u>Variable</u>	<u>Description</u>
IROWS	Row number identifying the element for stress output.
ICOLS	Column number identifying the element for stress output.

2.1.3 ADDITIONAL INPUT FILE

In addition to TAPE43, a separate user prepared input file called TAPE44 is required to control the execution of EAC and its output. The input data consist of a sequence of free field format input records, one record corresponding to each FORTRAN input list. A free field format input record contains a number of numerical data fields, each separated by one or more blanks.

EAC-1 Case Title

The Case Title is read on one line that may contain any character data, not exceeding 80 characters in length.

EAC-2 Execution Type Selection Record

Quite often, users may wish to rerun EAC for the same plate unit but with output generated for a different set of points or for distinct points on the example shell unit. There is an option which permits repeated execution of EAC for the same plate using a TAPE45 file generated by a previous execution.

<u>Variable</u>	<u>Description</u>
IRERUN	0 - Begin new case. Read EAC-2A,B 1 - Repeat with existing TAPE45. Go to EAC-3

EAC-2A Orthotropic stiffness coefficient record 1

The user must input the stiffness coefficients for the corresponding plate unit. These coefficients are available in the STAGS1 output for each component unit of the shell structure, and the user should select the appropriate data. The required membrane stiffness coefficients are

Variables: A11, A12, A22, A66

EAC-2B Orthotropic stiffness coefficient record 2

The required bending stiffness coefficients are

Variables: D11, D12, D22, D66

EAC-3 Solution Summary Record

The discrete data input to EAC via TAPE43 are stored in arrays whose dimensions are based on the STAGS grid. The displacement data arrays correspond to the nodes of this grid, while the stress and moment resultant data arrays correspond to the mid-points of the rectangular elements. The dimensions of these data arrays are limited by the core memory limitations on the host computer. These limits have not been established as yet. At present, the dimensions of the input arrays are limited to a maximum of 50 rows and 50 columns.

EAC can generate results either at a point ($IPT > 0$) or over a grid ($IPT = 0$). In the latter case, the dimensions of the output arrays, consisting of displacements, normal stress and moment resultants, transverse shears and an error estimate, correspond to the number of nodes in the grid. Here again, the dimensions are limited to a maximum of 50 rows and 50 columns. If necessary, the user can access a copy of the EAC source code and increase these limits. Instructions on how to do this are provided in Section 2.

<u>Variable</u>	<u>Description</u>
IPT	0 - generate output results over user grid >0 - generate results at IPT points
MT	Number of terms in X for product solution of in-plane homogeneous equations. (See 3.6)
NT	Number of terms in Y for product solution of in-plane homogeneous equations. (See 3.6)
MP	Number of terms in X for product solution of transverse homogeneous equation. (See 3.5)
NQ	Number of terms in Y for product solution of transverse homogeneous equation. (See 3.5)

If IPT=0 then go to EAC-4.

If IPT>0 then read IPT EAC-3A cards, go to EAC-5.

Note: See section 3.5 for limits on MT,NT,MP,NQ.

EAC-3A Point Coordinate Data Record

If IPT > 0 then provide IPT records with the X and Y coordinates of points for output data. There is a limit of 10 records, i.e., IPT \leq 10.

<u>Variable</u>	<u>Description</u>
X	X coordinate of point
Y	Y coordinate of point

EAC-4 Output Grid Definition Record

<u>Variable</u>	<u>Description</u>
NXS	Number of grid points in X-direction
NYS	Number of grid points in Y-direction

EAC-5 Output Summary Record

Rudimentary output control is provided.

<u>Variable</u>	<u>Description</u>
IPRT	0 - No output printed. 1 - Output printed on TAPE6
IGENC	0 - Generalized coordinates not printed 1 - Print generalized coordinates

EAC-6 Initial Imperfection Record

The presence of initial imperfections are accounted for in EAC by an imperfection subroutine WIMPD. The imperfection displacements are of the form

$$\bar{W} = \sum_{J=1}^K \text{IMN}(J) \sin M(J)\pi X/A \sin N(J)\pi Y/B$$

where IMN, M and N are limited to 24 each.

<u>Variable</u>	<u>Description</u>
IWIMP	0 - No imperfections 1 - Imperfection function parameters specified in records EAC-6A to EAC-6D

EAC-6A Imperfection Number Record

<u>Variable</u>	<u>Description</u>
K	Number of imperfections

EAC-6B Imperfection Coefficient Record

<u>Variable</u>	<u>Description</u>
IMN	K imperfection coefficients

EAC-6C M(J) Data Record

<u>Variable</u>	<u>Description</u>
M(J)	K M(J) data entries

EAC-6D N(J) Data Record

<u>Variable</u>	<u>Description</u>
N(J)	K N(J) data entries

If IWIMP=0, the input data file is complete.

If IWIMP=1, then read records EAC-6A to EAC-6D.

See Appendix A for nomenclature and Section 3 for theory.

2.1.4 INPUT TO PLOTTING POSTPROCESSOR EACPL

EACPL is the contour plotting postprocessor for EAC. The contour information is generated by EAC and stored on TAPE46. EACPL uses these data to plot two dimensional contour plots of the input and output of EAC. For general runs, the plots generated are of

1. STAGS OUTPUT - Displacements, stress and moment resultants
2. EAC OUTPUT - Displacements, stress and moment resultants, transverse shear, and error in the transverse equilibrium equation.

For point analysis runs (IPT>0 on EAC-3 card above), no EAC output plot data are generated, and consequently these plots will not be generated. EACPL needs an additional input data file called TAPE47.

EACPL-1 Grid Size Data Record

<u>Variable</u>	<u>Description</u>
NRWS	Number of nodal points in the X-direction for the STAGS grid. (repeated from F-1 card in STAGS1 data)
NCLS	Number of nodal points in the Y-direction for the STAGS grid. (repeated from F-1 card in STAGS1 data)
NXS	Number of nodal points in the X-direction for the user (output) grid. (repeated from EAC-4 card above)
NYS	Number of nodal points in the Y-direction for the user (output) grid. (repeated from EAC-4 card above)
PSCALE	Scale factor (based on plate dimensions). A value greater than unity enlarges the plot and a value smaller than unity will scale down the plot.
ICON	0 - system will select the contour details based on plot data. 1 - contour details given by user on 12 EACPL-2 cards.

EACPL-2 Contour Limit Detail Record

The contour limit details are given for the 12 parameters plotted. These are in order: U, V, W, NX, NY, NXY, MX, MY, MXY, QX, QY, ERR.

<u>Variable</u>	<u>Description</u>
SLO	lower limit on the function to be plotted.
SHI	upper limit on the function to be plotted.
SINC	step size, such that No. of Contours ≤ 40 .

The plotting program will use the same limits for corresponding STAGS and EAC plots.

2.2 PROCEDURES

The STAGSC-1 code is currently available on the Cyber machines for users at NASA Langley Research Center. A new experimental version of the STAGS code is currently installed on a MicroVAX in Building 1148. Given below are sample procedure files and instructions for the execution of STAGS and EAC on both these machines. Users are cautioned that the procedure files given below are not complete and must be treated as examples only. Users on other systems must consult their systems manager to interpret these instructions. The RESOURCE identification of the MicroVAX system on MICOM is 85.

2.2.1 WORKING ON THE CYBER

A complete execution of the STAGSC-1 system would typically involve executing three separate but independent modules; namely STAGS1, STAGS2 and STAPL. STAGS1 reads all input data describing the problem, preprocesses this input, writes local files containing data to be used by STAGS2, the main computational module, and STAPL, the plotting postprocessor. STAGS2 does not read any user-prepared input data, so a separate record in the input data deck is not included for STAGS2. STAPL, however, does require user-prepared input data describing plots to be generated and, if executed during a run, requires a record in the input data deck. The user may opt to bypass STAPL and obtain the plots using EACPL, the plotting postprocessor for EAC. EACPL generates plots of the displacements, stress and moment resultants for both the STAGS discrete output and the EAC continuous approximations. The advantage of using EACPL rather than STAPL is that the plots for the discrete variables and the continuous variables will both agree in scale and contour intervals for ease of direct comparison. In addition, transverse shears and the error in the transverse equilibrium equation computed by EAC are plotted.

The fourth module, which is either POSTP or NPOSTPX, can be used following a previous STAGSC-1 execution. POSTP enables the user to print quantities not selected for output in the original execution, while NPOSTPX is required to be executed to create a TAPE43 file as input to EAC. In addition to the creation of TAPE43, NPOSTPX enables the user to print output data for the plate unit and the load step under consideration. A new set of procedure files are provided to access copies of the absolute binary files

of STAGS1, STAGS2, STAPL, and POSTP/NPOSTPX or to create new absolute binary files incorporating any user-written subroutines that may be required. The master procedure file, STAGS, is accessed from the user's control card record as follows:

```
GET,STAGS/UN=STAGS,ST=CPF.
```

This file contains seven procedures; four associated with standard STAGSC-1 executions and three associated with STAGSC-1 executions using user written subroutines. The four procedures associated with standard execution are LITTLE, BIG, POST, and TP. Both LITTLE and BIG set up local copies of the absolute binary files for STAGS1, STAGS2 and STAPL. LITTLE should be used for models having less than 3,000 degrees of freedom. To access one of these procedures requires either of the following control cards:

```
BEGIN(LITTLE,STAGS)
```

or

```
BEGIN(BIG,STAGS)
```

The procedure POST sets up local copies of the absolute binary files for STAGS1, POSTP, and STAPL. POSTP processing is performed using a local file named TAPE20 (TAPE22 previously) and creates a new local file TAPE22. STAPL postprocessing is performed using the local file TAPE22. POST is accessed as follows:

```
BEGIN(POST,STAGS)
```

If user-written subroutines are included, it is necessary to reload the STAGSC-1 processors using one of the following procedures: RLBIG or RLPOST. To access these procedures requires the following control cards:

```
BEGIN(RLBIG,STAGS,UWS=lfm)
```

or

```
BEGIN(RLPOST,STAGS,UWS=lfm)
```

where the selection of which procedure to use is determined by the user's application and lfm is the name of a local file containing the FORTRAN-77

source statements of the user-written subroutines. If these reside on a permanent file, the user must GET or ATTACH this file and specify the local name lfn. The RL... procedures will compile lfn and generate new local absolute binary files for STAGS1, STAGS2 and POSTP.

In order to use POSTP/NPOSTPX, TAPE22 from a previous STAGS execution must be available as a local file named TAPE20. In addition, the user must have prepared and executed STAGS1 with the same model that produced the original TAPE22. Model data must be unaltered, but the print control flags and IPOST0, IPOST1, and IPOST2 may be changed as explained in the user's manual. It is not necessary to execute all modules in a single run.

In order to use EAC, TAPE43 generated by postprocessing with NPOSTPX must be available as a permanent file. Other data files needed are:

TAPE44	user input to control execution of EAC described above under the heading ADDITIONAL INPUT FILE.
TAPE45	file containing the solution in generalized coordinates, created by a prior EAC run.
TAPE46	input file for EACPL, generated by EAC if IRERUN=1 on EAC-2 card.(See ADDITIONAL INPUT)
TAPE47	input file to control execution of EACPL. (described as INPUT TO EACPL)

2.2.1.1 STANDARD BIG STAGSC-1 ANALYSIS

JOB...

USER...

CHARGE...

DELIVER...

FILESET,IA=CPF,DA=LPF.

GET,STAGS/UN=STAGS,ST=CPF.

BEGIN(BIG,STAGS)

STAGS1.

STAGS2.

REPLACE(TAPE22=pfn) (* pfn is the permanent file name of TAPE22 *)

STAPL.

PLOT.VARIAN,FF

E-O-R

STAGS1 analysis data

E-O-R

STAPL plotting data

E-O-F

2.2.1.2 STANDARD POST-PROCESSING

This procedure file is given for reference only. The procedure file to execute NPOSTPX to generate TAPE43 for EAC is given next.

```
JOB...
USER...
CHARGE...
DELIVER...
FILESET,IA=CPF,DA=LPF.
GET,STAGS/UN=STAGS,ST=CPF.
GET,TAPE20=pfn.
BEGIN(POST,STAGS)
STAGS1.
POSTP.
STAPL.
PLOT.VARIAN,FF
E-O-R
    STAGS1 data
E-O-R
    POSTP data
E-O-R
    STAPL data
E-O-F
```

2.2.1.3 POST-PROCESSING FOR GENERATION OF TAPE43

This is a skeleton procedure to execute NPOSTPX on the Cyber. It requires a TAPE20 file which was previously saved as TAPE22 from a STAGS-C1 execution. TAPE43 is required to analyse the STAGS results.

```
JOB...
USER...
CHARGE...
DELIVER...
FILESET,IA=CPF,DA=LPF.
GET,STAGS/UN=STAGS,ST=CPF.
GET,NPOSTPX/UN=STAGS,ST=CPF.
GET,TAPE20=pfn1. (* Where pfn1 is a previously saved TAPE22 *)
BEGIN(POST,STAGS)
STAGS1.
NPOSTPX.
REPLACE,TAPE43=pfn2.
E-O-R
    STAGS1 analysis data
E-O-R
    NPOSTPX data
E-O-F
```

2.2.1.4 STANDARD EAC EXECUTION

JOB...
USER...
CHARGE...
DELIVER...
FILESET, IA=CPF, DA=LPF.
GET, EAC, EACPL/UN=STAGS, ST=CPF.
GET, TAPE43=pfn1.
GET, TAPE44=pfn2.
GET, TAPE45=pfn3. (* ONLY FOR A RERUN *)
GET, TAPE47=pfn4.
EAC, TAPE43, TAPE6.
REWIND, TAPE6, TAPE46.
REPLACE, TAPE6=....
REPLACE, TAPE46=....
COPYSBF, TAPE43. (* PRINT DATA FILE *)
COPYSBF, TAPE6. (* PRINT OUTPUT FILE *)
EACPL.
PLOT.VARIAN, FF
E-O-F

2.2.1.5 STANDARD BIG STAGSC-1 ANALYSIS

If user-written subroutines are included then the following control cards will be required.

JOB...
USER...
CHARGE...
DELIVER...
GET, STAGS/UN=STAGS, ST=CPF.
BEGIN(RLBIG, STAGS, UWS=INPUT)
STAGS1.
STAGS2.
STAPL.
E-O-R

 User-written subroutines

E-O-R
 STAGS1 data
E-O-R
 STAPL data
E-O-F

The user-written subroutines could be in the file 'pfn', in which case use:

 BEGIN(RLBIG, STAGS, UWS=pfn)
and delete the statements:
 E-O-R
 user-written subroutines

2.2.1.6 MODIFYING EAC FOR LARGE GRIDS

At the present time, the input and output array dimensions are limited to a maximum of 50 rows and 50 columns each. To handle larger arrays, the user can access and modify a copy of the EAC and EACPL source codes. These are accessed by using the control cards:

```
GET,EACNOW/UN=STAGS,ST=CPF.  
GET,EACPLOT/UN=STAGS,ST=CPF.
```

To modify the limits on the INPUT (STAGS) arrays (determined by the F-2 card input in STAGS1), edit the source file and modify the values of SNX and SNY in the PARAMETER statements. SNX and SNY correspond to NRWS and NCLS respectively.

To modify the limits on the OUTPUT (USER) arrays, modify the values of NXU and NYU in the PARAMETER statements. For either the STAGS output arrays or the EAC output arrays, edit and modify the limits in EACPLOT. To compile and load, use the following procedure file:

```
JOB...  
USER...  
CHARGE...  
DELIVER...  
FILESET,IA=CPF,DA=LPF.  
GET,EACNOW,EACPLOT.  
GET,PLOT3D,GRAFLIB/UN=STAGS,ST=CPF.  
FTN5,I=EACNOW,L=MINE,LO=S,DB=PMD.  
ATTACH,FTN5MLB/UN=LIBRARY.  
LDSET,LIB=FTN5MLB,MAP=SBEX/MINE.  
LOAD,LGO  
NOGO,EAC  
RETURN,LGO.  
FTN5,I=EACPLOT,L=MINE,LO=S,DB=PMD.  
ATTACH,FTN5MLB,LRCGOSF/UN=LIBRARY.  
LDSET,LIB=FTN5MLB/PLOT3D/GRAFLIB/LRCGOSF,MAP=SBEX/MINE.  
LOAD,LGO.
```

NOGO, EACPL.

REPLACE, EAC, EACPL.

ENQUIRE.

EAC and EACPL are the new executables.

NOTE: Cyber has limited resident core memory, which restricts the field length available to the user. What this means is that there is a finite upper limit to which the grid size can be increased. This limit has not been determined.

2.2.2 WORKING ON THE MicroVAX

An experimental version of the STAGSC-1 computer code was installed on a MicroVAX in Building 1148 during May 1987. Currently, this version of STAGS does not have the STAPL postprocessor installed. POSTP is available and the VAX version will write a TAPE43 file whether the user wants to or not.

A copy of the error analysis code (EAC) has also been installed on the MicroVAX. System symbols have been defined to run STAGS and EAC on the MicroVAX and detailed instructions are given in the following pages.

2.2.3 EXECUTING STAGS ON THE VAX

The user is reminded that this is an experimental version of the STAGSC-1 code and its capabilities and reliability are yet to be documented. In this version, an additional input file is required for the execution of STAGS2. To run STAGS, the user enters the command :

RUNSTAGS <CR>

Entering this symbol initiates a command file that will submit a STAGSC-1 analysis as a batch job. The user must use the following naming convention for the input files and resulting output and restart files.

data.inp	defines the STAGS1 input data
data.bin	defines the STAGS2 input data
data.inpp	defines the POSTP input data (NOTE: different from Cyber!)
data.sod	defines the solution data file
data.rst	defines the previous restart file
data.out1	defines the STAGS1 run output
data.out2	defines the STAGS2 output
data.out3	defines the RESTART output (different versions for restarts)
data.outpp	defines the POSTP run output
data.t43	defines the TAPE43 created by POSTP
data.dat	defines STAGS1 FOR002.dat file needed to run STAGS2 without having to rerun STAGS1 each time.

In the above statements, the file name 'data' is arbitrary and could be the Case Title. However, the file name extensions are not arbitrary and must be as defined above.

2.2.4 SAMPLE STAGS RUN ON THE VAX

Prior to execution, the user must have correctly prepared

'data'.inp - data file for STAGS1

and

'data'.bin - data file for STAGS2

```
user:          RUNSTAGS <CR>
system:   Enter Root File Name...
user:          'data' <CR>
system:   Print Log File? Y or N.
user:          Y for yes or N for no <CR>
system:   Run after 9PM? Y or N.
user:          Y for yes or N for no <CR>
system:   Is this a RESTART? Y or N.
user:          Y for restart or TP, N otherwise<CR>.
system:   Ok to submit STAGS run? Y or N.
user:          Y to submit or N to abort. <CR>
```

STAGS2 execution on the VAX requires a series of new input records. The Thurston step/Transformation Processor (Ref. 3) referred to here is explained in Section 4.

The total user input to the program comes in two parts:

1. Input to STAGS1. This input block defines the model. It is completely unchanged from an ordinary model definition with the nonlinear static option turned on (INDIC=3).
2. Input to STAGS2. This input file contains all necessary functions for a static, nonlinear analysis (including corotation and the Riks algorithm).

The new input records TP-1, TP-2 and TP-3 contain the same data as the STAGS1 records A-1, C-1, and D-1, respectively. Data in the new records override the STAGS1 input records. This has the advantage that multiple solutions with differing load strategies can be run with the same STAGS1 data. The present version of STAGS requires an additional record TP-4.

TP-4 Secondary Path Parameter Record (see page 43 for parameter list)

This record is used to select either the Thurston step/Transformation processor (TP) or the Riks algorithm, and when selecting TP, to specify the type of modal analysis desired and which mode to use for the shifted point. If NPATH is 0, STAGS2 will attempt to continue the solution along the current load path, just as would be done in an ordinary STAGS solution. If NPATH is 1, a Thurston step will be attempted, and additional data from this record will be used. NEV specifies the number of bifurcation solutions to be calculated. NEQ specifies the number of modes to be used in the Thurston Equivalence Transformation. At present, NEQ cannot be greater than 1. NSOL describes the particular action of STAGS2, accounting for multiple modes. For NSOL = 1, it is assumed that only one mode is active, and a Thurston step is attempted using this mode for the shifted point. For NSOL=0, the modal coefficient matrix is saved and the STAGS2 run is terminated. IE denotes the mode number to be selected in computing the shifted point.

2.2.5 SAMPLE POSTP RUN ON THE VAX

Prior to a run, the user must have correctly prepared

1. 'data'.inpp - data file for POSTP
2. 'data'.dat - data file created by a STAGS1 run
- and 3. 'data'.sod - solution data file from the corresponding STAGS run.

```
user:          RUNPOSTP <CR>
system:  Enter Root File Name...
user:          'data' <CR>
system:  Print Log File? Y or N.
user:          Y for yes or N for no <CR>
system:  Run after 9PM? Y or N.
user:          Y for yes or N for no <CR>
system:  Ok to submit POSTP run? Y or N.
user:          Y to submit or N to abort. <CR>
```

2.2.6 EXECUTING EAC ON THE MicroVAX

To run EAC on the VAX, the user enters the command

RUNEAC

Entering this symbol initiates a command file which will submit an EAC analysis as a batch job. The user may directly access the executables for interactive execution:

```
RUN DUA0:[STAGS]EAC.EXE
```

and/or

```
RUN DUA0:[STAGS]EACPL.EXE
```

The square brackets are needed!

The user must use the following naming convention for the input files and resulting output files:

data.t43	defines the discrete FEM data to be analyzed (TAPE43)
data.t44	defines the EAC execution data file. (TAPE44)described as the ADDITIONAL INPUT FILE in Section 2.
data.t45	defines the intermediate file to which the generalized co-ordinates are written (TAPE45).
data.t46	defines the data file for the plotting postprocessor EACPL and is created by EAC (TAPE46)
data.t47	defines the EACPL execution data file described in Section 2 (TAPE47).
data.out	defines the EAC solution file.
data.log	defines the log file for system responses.

Again, the file name 'data' is arbitrary, while the file name extensions are not arbitrary and must be defined as above. The corresponding naming convention on the Cyber is given in parenthesis. The file definitions are the same as for the Cyber version.

2.2.7 SAMPLE EAC RUN ON THE MicroVAX

Prior to the execution, the user must have valid data files as defined above.

```
user:          RUNEAC <CR>
system:   Enter Root File Name.
user:          'data' <CR>
system:   Print Log File? Y or N.
user:          Y for yes and N for no <CR>
system:   Run after 9 PM? Y or N.
user:          Y for yes and N for no <CR>
system:   Is this a RERUN? Y or N.
user:          Y for yes and N for no <CR>
system:   OK, Submitting EAC file.
```

The results are returned in data.out and the graphics file as METAFILE.DAT. Currently the plots can be viewed only on TEK4109 terminals and copies made on hard copy machines. Alternatively, the plots can be printed (without viewing) on a printer such as the Post Script printer. To view the plots (on TEK4109 terminals), the user enters:

```
SETDRV 409 <CR>
```

```
MTRSHARE <CR>
```

The system loads the metafile translator and displays a new prompt 'M>'

```
M >S MF 1 METAFILE (assign the graphics file)
```

```
M >D P 1 (to draw picture one)
```

Press the spacebar to erase picture and continue.

```
M >D P n (to draw the 'n'th picture)
```

```
M >q (to terminate session)
```

To make a hard copy, on TEK 4109 terminals, press the SCOPY button.

To print the contour plots on the local Post Script printer in Building 1148, enter:

```

SETDRV PST <CR>
MTRSHARE <CR>
M >S MF 1 METAFILE
M >D P FROM FIRST TO LAST (to draw all pictures)
M >Q (to terminate session)

```

This creates a data file called POSTS.DAT. The user may wish to rename this file. The command 'PSDI3000 fn' will submit this file as a batch job to the Post Script printer.

NOTE: These instructions are only valid for the MicroVAX in Building 1148. Other users should refer to their systems coordinator.

2.2.8 INTERACTIVE USAGE OF EAC ON THE MicroVAX

Occasionally, the user may wish to execute EAC interactively on the VAX. This would certainly be necessary if the user wishes to evaluate the displacements, stress and moment resultants and transverse shear on a point by point basis. Prior to an interactive session, the user must assign the various data files used, to the corresponding I/O units. The following entries will make the necessary assignments:

```

ASSIGN 'DATA'.T43 FOR043
ASSIGN 'DATA'.T44 FOR044
ASSIGN 'DATA'.T45 FOR045
ASSIGN 'DATA'.T46 FOR046
ASSIGN 'DATA'.T47 FOR047
ASSIGN 'DATA'.OUT FOR006

```

To execute, enter

```
RUN DUA0:[STAGS]EAC
```

The results will be written to data.out. To generate the graphics file, enter

```
RUN DUA0:[STAGS]EACPL
```

Do not do this for single point evaluation sessions.

2.2.9 MODIFYING EAC FOR LARGE GRIDS

At the present time, the input and output data array dimensions are limited to a maximum of 50 rows and 50 columns each. To handle larger arrays, the user will have to access and modify a copy of the EAC source

code. This is available in the STAGS directory as a set of FORTRAN files.

The command

```
COPY DUA0:[STAGS.EAC.FOR]*.* *.*
```

will copy the following files into the user's current directory:

EAC.FOR

NHM.FOR

TRIMCT.FOR

ERRANA.FOR

EACS.OLB

RUNEAC.COM

MAKE_EXEC.COM

EACPLOT.FOR

To modify the limits on the input (STAGS) grid, edit the FORTRAN files (files with FOR as the three letter extensions) and modify the values of SNX and SNY in the PARAMETER statements. SNX and SNY correspond to NRWS and NCLS respectively. To modify the limits on the output (user) grid, edit ERRANA.FOR and modify the values of NXU and NYU in the PARAMETER statements. In either case, EACPLOT.FOR must be modified. To make a new (local) executable, enter

```
@MAKE_EXEC
```

This command file will compile and load the files to generate a local file called EAC.EXE. NOTE: Remember that the grid size cannot be made larger than what the computer can handle (memory limitations).

SECTION 3: THEORY

This section contains the equations solved by the numerical analysis in EAC. The equations serve to define the problem actually solved by the analysis and to aid in comparing the output from EAC to the discrete finite-element results from STAGS. The main output is a set of generalized coordinates that allow computing a continuous solution for the plate equations. The continuous solution has the obvious advantage that stress resultants and deflections can be computed at any point on the plate and is not restricted to grid points in STAGS. However, the main advantage of the continuous solution is that a measure of its error can be computed. The output of the code includes this measure of the error in the continuous solution, which is also an indirect measure of the error in the finite element results in STAGS.

In order to compute the continuous solution for a given plate from a shell structure, it is necessary to make some assumptions about the boundary conditions. In Ref. 4, the correction of the continuous solution by Newton's method is outlined along with removing the assumptions on the boundary conditions. The error measure computed in EAC is the starting point for this analysis, but correction of the results is beyond the scope of the analysis in the current version of EAC.

3.1 CONSTITUTIVE RELATIONS

The theory applies to orthotropic plates with the following constitutive relations:

$$N_x = A_{11} e_x + A_{12} e_y \quad (3.1a)$$

$$N_y = A_{12} e_x + A_{22} e_y \quad (3.1b)$$

$$N_{xy} = A_{66} e_{xy} \quad (3.1c)$$

$$M_x = D_{11} W_{,xx} + D_{12} W_{,yy} \quad (3.2a)$$

$$M_y = D_{12} W_{,xx} + D_{22} W_{,yy} \quad (3.2b)$$

$$M_{xy} = 2D_{66} W_{,xy} \quad (3.2c)$$

where the strain-displacement relations for the imperfect plate are

$$e_x = U_{,x} + \frac{1}{2} W_{,x}^2 + W_{,x} \bar{W}_{,x} \quad (3.3a)$$

$$e_y = V_{,y} + \frac{1}{2} W_{,y}^2 + W_{,y} \bar{W}_{,y} \quad (3.3b)$$

$$e_{xy} = U_{,y} + V_{,x} + W_{,x} W_{,y} + W_{,x} \bar{W}_{,y} + W_{,y} \bar{W}_{,x} \quad (3.3c)$$

3.2 EQUILIBRIUM EQUATIONS

The equations that determine the transverse shear stress resultants are moment equilibrium equations

$$Q_x = M_{x,x} + M_{xy,y} \quad (3.4a)$$

$$Q_y = M_{y,y} + M_{xy,x} \quad (3.4b)$$

The in-plane equilibrium equations are

$$N_{x,x} + N_{xy,y} = 0 \quad (3.5a)$$

$$N_{xy,x} + N_{y,y} = 0 \quad (3.5b)$$

The transverse force equilibrium equation is

$$Q_{x,x} + Q_{y,y} - N_3(F, W + \bar{W}) = 0 \quad (3.5c)$$

The notation $N_3(F, W + \bar{W})$ is a shorthand notation

$$N_3(F, W + \bar{W}) = N_3(F, W) + N_3(F, \bar{W}) \quad (3.6a)$$

$$N_3(F, W) = N_x W_{,xx} + 2N_{xy} W_{,xy} + N_y W_{,yy} \quad (3.6b)$$

3.3 DISPLACEMENT EQUATIONS AND BOUNDARY CONDITIONS

The constitutive relations, strain-displacement relations, and moment equilibrium equations are substituted to reduce the three force equilibrium

equations, equations (3.5), to three partial differential equations in the three displacement components U, V, and W:

$$L_{11}(U) + L_{12}(V) + N_1(W, W) + N_1(W, \bar{W}) + N_1(\bar{W}, W) = 0 \quad (3.7a)$$

$$L_{12}(U) + L_{22}(V) + N_2(W, W) + N_2(W, \bar{W}) + N_2(\bar{W}, W) = 0 \quad (3.7b)$$

$$L_{33}(W) - N_3(F, W) - N_3(F, \bar{W}) = 0 \quad (3.7c)$$

where

$$L_{11}(U) = A_{11} U_{,xx} + A_{66} U_{,yy} \quad (3.8a)$$

$$L_{12}(f) = (A_{12} + A_{66}) f_{,xy} \quad (3.8b)$$

$$L_{22}(V) = A_{22} V_{,yy} + A_{66} V_{,xx} \quad (3.8c)$$

$$L_{33}(W) = D_{11} W_{,xxxx} + 2(D_{12} + 2D_{66}) W_{,xxyy} + D_{22} W_{,yyyy} \quad (3.8d)$$

$$N_1(f, g) = A_{11} f_{,x} g_{,xx} + (A_{12} + A_{66}) f_{,y} g_{,xy} + A_{66} f_{,x} g_{,yy} \quad (3.8e)$$

$$N_2(f, g) = A_{22} f_{,y} g_{,yy} + (A_{12} + A_{66}) f_{,x} g_{,xy} + A_{66} f_{,y} g_{,xx} \quad (3.8f)$$

In the finite-element analysis in the STAGS code, the plate shell unit is usually part of a larger structure. The shell unit must satisfy geometric and natural boundary conditions and/or continuity conditions between edges of the plate and the rest of the shell structure. In the EAC code, the STAGS results for displacements on the plate boundary become the discrete boundary conditions for a continuous solution of the nonlinear partial differential equations in the dependent variables U, V, and W, equations (3.7).

3.4 CONTINUOUS SOLUTION FOR W

The continuous solution for W consists of two series solutions, W_E and W_I , with generalized coordinates computed from the discrete STAGS data.

$$W = W_I + W_E \quad (3.9a)$$

The functions W_I and W_E are chosen to satisfy two partial differential equations,

$$L_{33}(W_I) = N_3(F_0, W_0) + N_3(F_0, \bar{W}) \quad (3.9b)$$

$$L_{33}(W_E) = 0 \quad (3.9c)$$

The zero subscript on the right side of equation (3.9b) denotes discrete STAGS data. The equation is only satisfied at the centroids of the rectangular elements as defined by the STAGS1 input on the number of rows and columns. The continuous solution for W_I is computed by first using numerical harmonic analysis, Ref. 5, to compute the coefficients N_{3mn} in a double sine series passing through the discrete values on the right-hand side of equation (3.9b),

$$N_3(F_0, W_0 + \bar{W}) = \sum_{m=1}^{M-1} \sum_{n=1}^{N-1} N_{3mn} \sin m\pi x/a \sin n\pi y/b \quad (3.10a)$$

The number of terms in the series is determined by M and N, the number of rows and columns in the STAGS1 model. The generalized coordinates in the continuous solution W_I are the Fourier coefficients A_{mn} in the series

$$W_I = \sum_{m=1}^{M-1} \sum_{n=1}^{N-1} A_{mn} \sin m\pi x/a \sin n\pi y/b \quad (3.10b)$$

Substituting equations (3.10) into equation (3.9b) determines the numerical values of the A_{mn} .

$$A_{mn} = \frac{a^4}{\pi^4} \frac{N_{3mn}}{[D_{11}m^4 + 2(D_{12} + 2D_{66})m^2n^2\beta^2 + D_{22}n^4\beta^4]} \quad (3.10c)$$

In general, the continuous solution W_I does not satisfy the plate boundary and/or continuity conditions. The solution W_E is added to W_I to obtain a better fit of the discrete results on the boundary. The solution W_E is assumed as a series that satisfies equation (3.9c) term by term.

$$W_E = W_{00} + \sum_{p=1}^P [W_{pc}(\eta) \cos p\xi + W_{ps}(\eta) \sin p\xi] \\ + \sum_{q=1}^Q [W_{qc}(\xi) \cos q\eta + W_{qs}(\xi) \sin q\eta] \quad (3.11)$$

The first term is a polynomial with undetermined coefficients

$$W_{00} = a_{00} + a_{10} x + a_{01} y + a_{20} x^2 + a_{02} y^2 \quad (3.12)$$

Each of the functions $W_{pc}(\eta)$, $W_{ps}(\eta)$, $W_{qc}(\xi)$, and $W_{qs}(\xi)$ satisfies a fourth order, homogeneous, ordinary differential equation and, consequently, contains four constants of integration that make up the set of generalized coordinates for W_E .

3.5 BOUNDARY CONDITIONS FOR W

The undetermined coefficients in W_{00} and the constants of integration in the functions $W_{pc}(\eta)$, $W_{ps}(\eta)$, $W_{qc}(\xi)$, and $W_{qs}(\xi)$ are computed from discrete STAGS output on the plate boundary. The output for W , $W_{,x}$ on sides with constant x , and $W_{,y}$ on sides with constant y at nodal points become the discrete boundary conditions for the continuous solution for W .

Accepting the STAGS results as correct on the plate boundary is an assumption in deriving the continuous solution in EAC. After all the generalized coordinates in the continuous solution for U, V , and W are computed, the effects of the assumption can be examined. However, correction of errors on the boundary of the plate is beyond the scope of the theory programmed in EAC.

The constants of integration in the product solutions in W_E are computed by a least-squares analysis of the equations,

$$W_E = W_0 - W_I \quad x=0 \text{ and } x=a \quad (3.13a,b)$$

$$W_{E,x} = W_{0,x} - W_{I,x} \quad x=0 \text{ and } x=a \quad (3.13c,d)$$

$$W_E = W_0 - W_I \quad y=0 \text{ and } y=b \quad (3.14a,b)$$

$$W_{E,y} = W_{0,y} - W_{I,y} \quad y=0 \text{ and } y=b \quad (3.14c,d)$$

For example, four equations containing the four constants of integration appearing in the function $W_{rc}(\eta)$, where r is an integer, are generated by multiplying each of the equations (3.13) by $\cos r\eta$ and integrating from 0 to 2π . Terms in $W_{qc}(\eta)$ for $q \neq r$ drop out of the four integrals along with all the functions $W_{qs}(\eta)$. However, the integrals determine nonzero coefficients multiplying the constants of integration appearing in the functions $W_{pc}(\eta)$ and $W_{ps}(\eta)$.

In a corresponding fashion, four integrals are written for each trigonometric function $\cos q\eta$, $\sin q\eta$, $\cos p\xi$, and $\sin p\xi$ appearing in W_E . Integrals are also written containing nonzero multipliers of the undetermined coefficients in W_{00} . Four of these latter integrals are integrals of equations (3.13c), (3.13d), (3.14c) and (3.14d). An additional equation containing the coefficient a_{00} is derived by summing the four integrals of equations (3.13a), (3.13b), (3.14a) and (3.14b).

The integrals derived from the left-hand sides of equations (3.13) and equations (3.14) are equated to corresponding integrals of the right hand sides. Since the notation W_0 refers to discrete STAGS data at nodal points, the integrals containing W_0 and its partial derivatives are replaced by numerical quadrature formulas.

In order for the numerical quadrature formulas to be meaningful, there is an upper limit $p=P$ and $q=Q$ for the number of terms in the series for W_E , equation (3.11). The upper limits P and Q must satisfy the inequalities (see ref. 5)

$$2P + 1 \leq M \quad (3.15a)$$

$$2Q + 1 \leq N \quad (3.15b)$$

where M and N are the number of rows and columns in the STAGS data. The number of terms, P and Q , correspond to the FORTRAN variables, MP and NQ on card EAC-3 in the file TAPE44, and M and N are $NROWS$ and $NCOLS$ in the input file for STAGS1.

The integrations defined above result in a set of linear algebraic equations for a least-squares fit of the STAGS boundary data on W . The equations are solved to compute numerical values for $(8P+8Q)$ constants of integration and for five undetermined coefficients in the polynomial W_{00} . In the computations that follow in EAC, the continuous function $W=W_E+W_I$ is used wherever W or its partial derivatives appear in the equations.

3.6 CONTINUOUS SOLUTIONS FOR U AND V

The continuous solutions for U and V are computed following the same general plan as the solution for W . A double Fourier series is used to generate solutions U_I and V_I of the in-plane equilibrium equations, equations (3.7a,b). Solutions of

$$L_{11}(U_E) + L_{12}(V_E) = 0 \quad (3.16a)$$

$$L_{12}(U_E) + L_{22}(V_E) = 0 \quad (3.16b)$$

are added so that continuous solutions of U and V are of the form

$$U = U_I + U_E \quad (3.17a)$$

$$V = V_I + V_E \quad (3.17b)$$

Constants of integration appearing in the solutions U_E and V_E are computed from a least-squares fit of discrete STAGS data for values of U and V at node points.

The specific forms for U and V are

$$U_I = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} A_{1mn} \cos m\xi \cos n\eta + \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} B_{1mn} \sin m\xi \sin n\eta \\ + \sum_{m=0}^{\infty} \sum_{n=1}^{\infty} C_{1mn} \cos m\xi \sin n\eta + \sum_{m=1}^{\infty} \sum_{n=0}^{\infty} D_{1mn} \sin m\xi \cos n\eta \quad (3.18a)$$

$$V_I = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} A_{2mn} \cos m\xi \cos n\eta + \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} B_{2mn} \sin m\xi \sin n\eta \\ + \sum_{m=0}^{\infty} \sum_{n=1}^{\infty} C_{2mn} \cos m\xi \sin n\eta + \sum_{m=1}^{\infty} \sum_{n=0}^{\infty} D_{2mn} \sin m\xi \cos n\eta \quad (3.18b)$$

$$U_E = a_1 + b_1 x + c_1 y + \sum_{p=1}^{\infty} [F_{1p}(\eta) \cos p\xi + G_{1p}(\eta) \sin p\xi] \\ + \sum_{q=1}^{\infty} [H_{1q}(\xi) \cos q\eta + I_{1q}(\xi) \sin q\eta] \quad (3.19a)$$

$$V_E = a_2 + b_2 x + c_2 y + \sum_{p=1}^{\infty} [F_{2p}(\eta) \cos p\xi + G_{2p}(\eta) \sin p\xi] \\ + \sum_{q=1}^{\infty} [H_{2q}(\xi) \cos q\eta + I_{2q}(\xi) \sin q\eta] \quad (3.19b)$$

For solution of equations (3.16), the functions $F_{1p}(\eta)$ and $F_{2p}(\eta)$ satisfy the pair of ordinary differential equations

$$A_{66} \beta^2 F_{1p}'' - p^2 A_{11} F_{1p} + (A_{12} + A_{66}) \beta p F_{2p}' = 0 \quad (3.20a)$$

$$- (A_{12} + A_{66}) \beta p F_{1p}' + A_{22} \beta^2 F_{2p}'' - p^2 A_{66} F_{2p} = 0 \quad (3.20b)$$

where primes denote differentiation with respect to η . The subscripted functions G , H , and I satisfy similar coupled ordinary differential equations. The complete solution of equations (3.20) contains four constants of integration. The constants of integration are determined by boundary conditions on U and V .

$$U_E = U_0 - U_I \quad x=0 \text{ and } x=a \quad (3.21a,b)$$

$$V_E = V_0 - V_I \quad x=0 \text{ and } x=a \quad (3.21c,d)$$

$$U_E = U_0 - U_I \quad y=0 \text{ and } y=b \quad (3.22a,b)$$

$$V_E = V_0 - V_I \quad y=0 \text{ and } y=b \quad (3.22c,d)$$

The solution for the undetermined coefficients and constants of integration in U_E and V_E is analogous to the solution for W_E . In particular, the number of terms in the series solutions must satisfy the inequalities in equations (3.15). The FORTRAN variables MT and NT on card EAC-3 in the file TAPE44 correspond to the number of p and q terms in U_E and V_E . The upper limits on p and q must again satisfy the inequalities of equations 3.15.

3.7 OUTPUT AND ERROR ANALYSIS

The least-squares solution for the generalized coordinates in U_E and V_E completes the analysis to determine a continuous solution for the plate displacements U , V , and W . The stress resultants N_x , N_y , N_{xy} , M_x , M_y , M_{xy} , Q_x , and Q_y are then computed from the displacements and their derivatives. The output from the continuous solution can be computed at any given point or over a rectangular grid (see EAC cards in NPOSTPX).

A measure of the error in the continuous solution derived from the finite element results is $E_3(x,y)$, the residual error in satisfying the transverse equilibrium equation, equation (3.7c).

$$E_3 = L_{33}(W) - N_3(F, W + \bar{W}) \quad (3.23)$$

where W is the continuous solution defined by equations (3.9) and the in-plane stress resultants appearing in the bilinear operator are computed from the continuous solution. The output from EAC normalizes E_3 by dividing the residual error by the maximum absolute value computed for the linear operator $L_{33}(W)$.

Away from the plate boundary, the error is a good measure of the accuracy of the results from the continuous solution and represents a result that is not available in a finite-element solution based on direct minimization of a variational principle. Near the plate boundary, the error

contains a "Gibb's phenomenon" because of using a double sine series for W_I and the imperfection \bar{W} . On the boundary

$$L_{33}(W_I) = 0$$

and

$$E_3 = -N_3(F, W_E)$$

while away from the boundary the general error, from equations (3.9), is

$$E_3 = N_3(F_0, W_0 + \bar{W}) - N_3(F, W + \bar{W}) \quad (3.24)$$

On the plate boundary, other measures exist for examining the accuracy of the continuous solution. On boundaries, where STAGS1 specifies constant displacements, the output from EAC can be compared directly to the STAGS output at nodal points. When the maximum number of linearly independent generalized coordinates defined in equations (3.15) is used in the continuous solution, the least-squares results for displacements will agree with the STAGS results at nodal points on the boundary except at the four corners of the plate. The continuous solution can also be checked against displacement boundary conditions between nodes.

In addition to displacements, stress resultants are also computed on the boundary in EAC. This allows checking the error in the stress resultants. For example, on a simply-supported edge perpendicular to the x-direction, the bending moment resultant, M_x , should vanish. The analysis in EAC matches the rotation, $W_{,x}$, on this boundary and the residual M_x on a simply-supported edge is a measure of error although it will not be clear whether the error is in the continuous solution or in the edge rotations from the finite-element solution.

When continuity relations are specified for the edge of a plate at a juncture with other shell units, the accuracy of the results from STAGS and EAC are more difficult to assess. The STAGS results will satisfy continuity of displacements at the common nodes of the shell units at a juncture. The

continuous solution can also be used to check continuity between nodes along a common boundary.

When a number of shell units make up a statically indeterminate structure, it is of interest to determine the load paths in the structure. EAC computes output for in-plane force resultants on the edges, in the x-direction

$$F_{x1} = - \int_0^b N_x dy \quad x=0 \quad (3.25a)$$

$$F_{x2} = - \int_0^a N_{xy} dx \quad y=b \quad (3.25b)$$

$$F_{x3} = - \int_0^b N_x dy \quad x=a \quad (3.25c)$$

$$F_{x4} = - \int_0^a N_{xy} dx \quad y=0 \quad (3.25d)$$

and the y-direction,

$$F_{y1} = - \int_0^b N_{xy} dy \quad x=0 \quad (3.26a)$$

$$F_{y2} = - \int_0^a N_y dx \quad y=b \quad (3.26b)$$

$$F_{y3} = - \int_0^b N_{xy} dy \quad x=a \quad (3.26c)$$

$$F_{y4} = - \int_0^a N_y dx \quad y=0 \quad (3.26d)$$

SECTION 4: TP USER'S INSTRUCTIONS

A new processor called TP (transformation processor) is available. TP allows the user to obtain a nonlinear solution on an equilibrium path different from the primary equilibrium path. This processor examines the solution on the primary path and for a fixed set of input parameters will compute solutions on another path, well into the postbuckling range. Ref. (2) outlines the numerical analysis for the TP processor and contains two example cases.

The Bifurcation Point Processor, TP, is an executable program which replaces STAGS2 in the normal execution sequence (STAGS1, STAGS2). TP can perform any of the following steps in one execution:

1. Nonlinear collapse analysis from any specified restart solution to a specified maximum load level. The restart solution may be on a primary or a secondary path. The Riks path length method is used here. This step is the same as a normal STAGS2 execution using the Riks method.

2. Nonlinear collapse analysis as in (1) above with the additional computation of the closest bifurcation modes whenever a matrix factorization is performed. The bifurcation modes are saved on the restart file TAPE22 and are thus available in later execution steps as initial estimates or for use in step 3.

3. Secondary solution computations using Thurston's method. This type of execution can be performed as a separate step or can sometimes be preceded with computations described in (2) above.

TP requires a short input file to control execution of the different options available to the user. The contents of this data file and the execution of TP differ depending on the computer, Cyber or MicroVAX.

4.1 WORKING ON THE CYBER

The input data are read in free format as described for the STAGS1 input file.

TP-1 Case Title

The case title may contain any text up to 72 characters.

<u>Variable</u>	<u>Description</u>
TITLE	Case title

TP-2 Load Multiplier Record.

This record is the same as STAGSC-1 record C-1.

<u>Variable</u>	<u>Description</u>
STLD1	Starting load factor for system A.
STEP1	Load factor increment for system A.
FACM1	Maximum load factor for system A.
STLD2	Starting load factor for system B.
STEP2	Load factor increment for system B.
FACM2	Maximum load factor for system B.
ITEMP	0 - No thermal loads. 1 - Temperatures referred to system A. 2 - Temperatures referred to system B.
INERT	0 - No gravitational/centrifugal effects. 1 - Grav/cent loads in system A. 2 - Grav/cent loads in system B.

TP-3 Strategey Parameter Record.

This record is the same as the STAGSC-1 record D-1.

<u>Variable</u>	<u>Description</u>
ISTART	0 - Begin new case. N - Restart from Nth load step.
NSEC	No. of CPU seconds of execution (approx).
NCUT	Total no. of times the step size may be cut.
NEWT	Total no. of refactorings allowed. NEWT < 0 Refactor every NEWT load steps. NEWT=-20 True Newton method.
NSTRAT	0 - Initial estimate from extrapolation. 1 - Previous soln. used as initial estimate.

	-1 - Path length used as independent param.
DELX	Error tolerance. (if DELEX=0, default DELEX=.001 is used)
WUND	Relaxation factor. (if WUND=0, the program chooses WUND)

TP-4 Secondary Path Parameter Record.

The parameters in this record control the branching from the primary path to a secondary path. After solutions on a secondary path have been obtained and saved on the restart file, the analysis may be continued with the standard "primary" path method using a secondary path solution as the restart load step.

<u>Variable</u>	<u>Description</u>
NPATH	0-Primary path (Riks' method). 1-Secondary path (Thurston's method).
NEV	0-No bifurcation solutions will be found. 1-Find N bifurcation modes.
NEQ	0-Do not continue to bifurcation path. N-Equivalence transformation using N modes. (NOTE: NEQ cannot exceed NEV)
NSOL	0-Save reduced system on TAPE31; stop 1-Save reduced system on TAPE31; solve for load factor PA with QFAK=0 using equivalence transformation. -1-Read PA, QFAK (TP-6); obtain solution using equivalence transformation.
IE	Number of bifurcation mode used to compute shifted point. (Not used if NPATH or NEQ = 0.)

TP-5 Secondary Solution Path Reduced System Record.

This record is included only if NPATH=1 and NEQ>0 on the TP-4 record. STEP specifies the amplitude of the eigenvector of the tangent stiffness matrix chosen to produce a secondary solution. (The STAGS output labels these eigenvectors as "buckling modes"). The "first shifted point" is obtained by adding STEP times the IE'th mode to the last solution on the primary path.

The eigenvectors have been normalized so that the largest component is 1. DELTA determines the perturbations of the NEQ eigenvectors from the "first shifted point". A reduced polynomial system is thus obtained to provide for numerical solutions to the complete nonlinear problem.

<u>Variable</u>	<u>Description</u>
STEP	Amplitude of bifurcation mode IE used to construct the shifted point (i.e., $X0 = X + STEP * EV(IE)$).
DELTA	Coefficient of perturbations used for reduced system.

TP-6 Eigenvector Amplitude Record.

This record is included only if NSOL=-1 on the TP-4 record. The equivalence transformation determined by NEQ eigenvectors is used with the load factors PA and PB and modal amplitudes QFAK to obtain a solution in the full discrete system.

<u>Variable</u>	<u>Description</u>
PA	Load factor for system A for secondary solution.
PB	Load factor for system B for secondary solution.
QFAX(I) (I=1,NEQ)	Amplitudes of eigenvectors for secondary solution.

4.1.1 STANDARD TRANSFORMATION PROCESSING (TP)

JOB...
USER...
CHARGE...
DELIVER...
FILESET, IA=CPF, DA=LPF.
GET, STAGS/UN=STAGS, ST=CPF.
GET, TAPE20=pfn.
BEGIN(TP, STAGS)
STAGS1.
STAGS2.
STAPL.
PLOT.VARIAN, FF
E-O-R
 STAGS1 data
E-O-R
 TP data
E-O-R
 STAPL data
E-O-F

If user-written subroutines are included, it will be necessary to reload the STAGSC-1 processors using the procedure RLTP. This process is similar to that described in section 2.2.1 for RLBIG and RLPOST.

4.2 WORKING ON THE MicroVAX

On the MicroVAX, TP is still called STAGS2 and always requires input. The input to TP was described in Section 2 under the heading SAMPLE STAGS RUN ON THE VAX. To select and use the TP algorithm, three of the input cards are modified.

On TP-2 STEP is replaced by the value of the estimated modal amplitude.

On TP-3 NSTRAT must be set to -1 for the TP option to be used.

On TP-4 select NPATH to be 1. If NPATH = 1, a Thurston step will be attempted, and additional data from this record will be used. NEV specifies the number of eigenvectors to calculate. NEQ specifies the number of modes to be used in the Thurston Equivalence Transformation. NSOL must be selected as unity.

APPENDIX A: NOMENCLATURE

Analysis Variable	FORTTRAN Name	Description
$A_{11}, A_{12}, A_{22}, A_{66}$	A11,A12,A22,A66	plate extensional stiffnesses
A_{mn}, N_{3mn}		Fourier sine-sine series coefficients
a,b	A,B	dimensions of rectangular plate
$D_{11}, D_{12}, D_{22}, D_{66}$	D11,D12,D22,D66	plate bending stiffnesses
e_x, e_y, e_{xy}		in-plane strains
E	E3	residual error
f,g		dummy variables in linear operators
$L_{ij} () \ i=1,2; \ j=1,2$		linear operators
M_x, M_y, M_{xy}	MX,MY,MXY	moment stress resultants
N_x, N_y, N_{xy}	NX,NY,NXY	in-plane stress resultants
x,y	XX(I),YY(J)	Cartesian coordinates
U,V,W	U,V,W	plate displacements
\bar{W}	WI	initial out-of-plane imperfection
$\beta=a/b$	BETA	plate aspect ratio
$\xi=2\pi x/a, \ \eta=2\pi y/b$		dimensionless coordinates

APPENDIX B: EXAMPLE

A stiffened plate assembly (Ref. 6) is shown in Fig. 1. The general purpose finite element code STAGS is used to analyse this plate assembly in the postbuckled state. The plate assembly is simply supported on two edges and symmetric on the other two edges. A compressive load twice the critical load is applied on the simply supported edges. Data for this sample problem are given in Table 1. Assuming the finite element results are correct on the edges of the shell unit, the error analysis code EAC is used to obtain numerical results for shell unit 3, a plate that is bounded on two sides by a juncture with a stiffener and another plate shell unit. Sample data files are given here for the STAGS and the EAC runs on the Cyber and on the MicroVAX. Sample contour plots generated by EACPL are also included in Figs. 2-13.

Note: Currently EAC does not run with the corotation option.

WORKING ON THE CYBER

For a line by line explanation of the STAGS input data cards, the user is referred to the STAGS users manual (Ref. 1). These data cards indicate that STAGS is to use the non-corotational procedure and nonlinear analysis. There is a single loading system compressing the panel and analysis is to be done from a starting load factor of 0.4 to a final load factor of 1.4 in steps of 0.2. There are 5 shell units. The connections between the shell units are given on the G-1 cards and the partial compatibility constraints for uniform end shortening are given on the G-2 cards. Material and wall data are given on the I and K cards. Shell unit details are given on the M cards.

STAGS1 INPUT DATA

5 BRANCH STIFFENED PANEL MODEL

3 1 1 0 0 0 1 0 0 1 \$B-1

5 0 0 4 10 0 0 \$B-2

1 0 2 \$B-3

0.4 0.2 1.4 \$C-1
 0 5000 0 -20 0 1.0E-4 \$D-1
 17,5 17,3 17,7 17,3 17,5 \$F-1
 C
 C DEFINE CONNECTIONS BETWEEN SHELL UNITS
 C
 1 2 3 4 \$G-1
 1 2 2 4 \$G-1
 3 2 4 4 \$G-1
 3 2 5 4 \$G-1
 C
 C DEFINE PARTIAL COMPATIBILITY CONSTRAINTS
 C FOR UNIFORM END SHORTENING
 C
 3 1 3 1 1 1 0 1 \$G-2
 3 1 3 1 2 1 0 1 \$G-2
 3 1 3 1 3 1 0 1 \$G-2
 3 1 3 1 4 1 0 1 \$G-2
 3 1 3 1 5 1 0 1 \$G-2
 3 17 3 1 1 17 0 1 \$G-2
 3 17 3 1 2 17 0 1 \$G-2
 3 17 3 1 3 17 0 1 \$G-2
 3 17 3 1 4 17 0 1 \$G-2
 3 17 3 1 5 17 0 1 \$G-2
 C
 C MATERIAL AND WALL DATA
 C
 1 \$I-1
 1.32E7 0.3 4.54E6 0.101 0.0 9.8E6 0.0 \$I-2
 1 1 1 \$K-1
 1 0.084 \$K-2
 2 1 1 \$K-1
 1 0.058 \$K-2
 C
 C UNIT 1

C
2 3 \$M-1
0.0 15.0 0.0 2.5 \$M-2A
0.0 0.0 0.0 \$M-4
0.0 2.5 0.0 \$M-4
15.0 2.5 0.0 \$M-4
1 \$M-5
411 \$
1 6 1 4 \$P-1
0 0 0 0 \$Q-1
1 5 \$R-1

C

C UNIT 2

C

2 3 \$M-1
0. 15.0 0.0 1.352 \$M-2A
0.0 2.5 0.0 \$M-4
0.0 2.5 1.352 \$M-4
15.0 2.5 1.352 \$M-4
2 \$M-5
411 \$
1 6 1 6 \$P-1
0 0 0 0 \$Q-1
1 5 \$R-1

C

C UNIT 3

C

2 3 \$M-1
0.0 15.0 0.0 5.0 \$M-2A
0.0 2.5 0.0 \$M-4
0.0 7.5 0.0 \$M-4
15.0 7.5 0.0 \$M-4
1 \$M-5
411 \$
1 6 1 6 \$P-1
1 0 0 0 \$Q-1

```

1 3 $Q-2
8000.0 1 1 1 3 0 $Q-3
-8000.0 1 1 17 3 0 $Q-3
0.0 -1 1 9 3 0 $Q-3 ELIMINATE U RIGID BODY MOTION
1 5 $R-1
C
C UNIT 4
C
2 3 $M-1
0.0 15.0 0.0 1.352 $M-2A
0.0 7.5 0.0 $M-4
0.0 7.5 1.352 $M-4
15.0 7.5 1.352 $M-4
2 $M-5
411 $
1 6 1 6 $P-1
0 0 0 0 $Q-1
1 5 $R-1
C
C UNIT 5
C
2 3 $M-1
0.0 15.0 0.0 2.5 $M-2A
0.0 7.5 0.0 $M-4
0.0 10.0 0.0 $M-4
15.0 10.0 0.0 $M-4
1 $M-5
411 $
1 4 1 6 $P-1
0 0 0 0 $Q-1
1 5 $R-1

```

TP DATA FILE

Whereas STAGS2 does not normally read input data, TP reads a short input file to control the execution of the different options available to the user.

This input file also permits the user to change certain data records which may have been read in STAGS1. On TP-2, the initial and final load factors are specified as the load factor at which TP is to be attempted. The data below are representative.

5 BRANCH STIFFENED PANEL MODEL. TP RUN \$TP-1

1.0 0.1 1.0 \$TP-2

4 8000 2 -20 1 0 \$TP-3

1 2 1 1 1 \$TP-4

0.04 0.1 \$TP-5

DATA FOR RESTART FROM TP

After solutions on a secondary path have been obtained and saved on the restart file, the analysis may be continued with the standard "primary" path method using a secondary path solution as the restart load step. In the STAGS1 data file, modify the C-1 and the D-1 records.

1.204 0.2 2.0 \$C-1

5 8000 0 -20 1 1.0E-4 \$D-1

WORKING ON THE MicroVAX

STAGS1 DATA

Same as for Cyber.

STAGS2 DATA

There is no corresponding data file on the Cyber. The first three input records TP-1, TP-2 and TP-3 contain the same data as the STAGS1 A-1, C-1, and D-1 records, respectively. Data in the new records override the STAGS1 input records.

5 BRANCH STIFFENED PANEL MODEL \$TP-1

1.0 0.1 1.0 \$TP-2

0 8000 8 8 0 1.0E-4 \$TP-3

1 \$TP-4

TP DATA

When the Thurston Equivalence Transformation is to be used, the data item STEP on record TP-2, now takes the value of the modal amplitude. The value of NSTRAT on TP-3 is -1. The value of NPATH on TP-4 is 1 and additional data from this record will be used. In the VAX version, STAGS will continue the analysis along the secondary path till a desired load factor (as specified in item 3 of TP-2 card) is achieved. On the Cyber, the user has to restart from the load factor to which TP converges.

```
5 BRANCH STIFFENED PANEL MODEL $TP-1
1.0 0.04 2.0 $TP-2
1 8000 2 -20 -1 0 $TP-3
1 2 1 1 2 $TP-4
```

The modal amplitude has been specified to be 0.04; the equivalence transformation uses 2 vectors with eigenvector 2 as the initial direction. Two eigenvectors (denoted as "buckling modes" in the STAGS output) will be computed. Start analysis from step 1 of the previous run. Continue the analysis along the selected path to a load factor of 2.0.

RESTARTS

A STAGSC-1 analysis can be restarted from any load step for which the solution has been saved on a restart file during a previous execution.

Example:

```
5 BRANCH STIFFENED PANEL MODEL $TP-1
1.220 0.4 2.622 $TP-2
5 8000 0 -20 1 0.05 $TP-3
1 $TP-4
```

The analysis is restarted from the 5th load step with a starting load factor of 1.22 and a final load factor of 2.622, with a load step of 0.4. The execution time is expected to be less than 8000 seconds.

EAC INPUT DATA

EAC is used to obtain numerical results for plate component 3 that is bounded on two sides by a juncture with a stiffener and another plate component. The input data indicate that this is a new case, that results are to be generated over a grid defined in the next record using 8 terms in X and 3 terms in Y for both the inplane and the transverse equations' homogeneous solutions. Printout of input data and output results is requested, and the analysis is for a plate with no defined initial geometric imperfections.

SHELL UNIT 3 OF 5 ELEMENT STIFFENED PANEL

```
0 $EAC-2
.126176E+07 .378527E+06 .936758E+06 .381360E+06 $EAC-2A
.741912E+03 .222574E+03 .550814E+03 .224240E+03 $EAC-2B
0 8 3 8 3 $EAC-3
33 13 $EAC-4
1 0 $EAC-5
0 $EAC-6
```

EACPL INPUT DATA

The following are the input data for the contour plotting postprocessor. The input grid is 17 by 7, the output grid is 33 by 13, the contour plots are to be scaled to 40% of the plate dimensions, and the contour limits are user specified in 12 EACPL-2 records. Note: When contour data are user input, all 12 EACPL-2 records must be present.

```
17 7 33 13 0.4 1 $EACPL-1
-0.01 0.01 0.001 $EACPL-2
-0.001 0.001 0.0001 $EACPL-2
-0.1 0.1 0.01 $EACPL-2
-2000.0 0.0 50.0 $EACPL-2
-500.0 500.0 50.0 $EACPL-2
-50.0 50.0 5.0 $EACPL-2
-30.0 30.0 2.0 $EACPL-2
-30.0 30.0 2.0 $EACPL-2
```


-30.0 30.0 2.0 \$EACPL-2
-80.0 76.0 4.0 \$EACPL-2
-80.0 76.0 4.0 \$EACPL-2
-0.02 0.019 0.001 \$EACPL-2

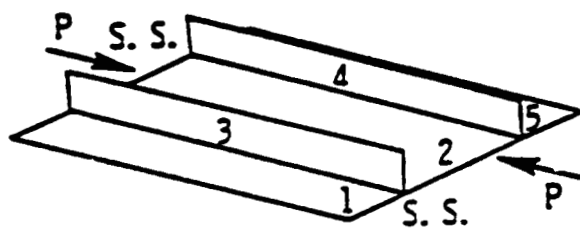
REFERENCES

1. Almroth, B. O.; Brogan, F. A.; and Stanley, G. M.: Structural Analysis of General Shells, Vol. II, User's Instructions for STAGSC-1. Lockheed Report LMSC-D633873, NASA CR-165671, January 1981.
2. Thurston, G. A.; Brogan, F. A.; and Stehlin, P.: Postbuckling Analysis Using a General Purpose Code. AIAA Journal, Vol. 24, No. 6, pp. 1013-1020 June, 1986.
3. Rankin, C. C.; Stehlin, P.; and Brogan, F. A.: Enhancements to the STAGS Computer Code. NASA Contractor Report 4000, November 1986.
4. Thurston, G. A.; Reissner, J. E.; Stein, P. A.; and Knight, N. F. Jr.: Error Analysis and Correction of Discrete Solutions from Finite Element Codes. AIAA Journal, Vol 26, No. 4, April, 1988.
5. Hildebrand, F. B.: Introduction To Numerical Methods. McGraw-Hill, New York, 1956.
6. Sistla, R.; and Thurston, G. A.: Error Analysis of Finite Element Solutions for Postbuckled Plates. Proceedings of the AIAA/ASME/ASCE/AHS/ASC 29th Structures, Structural Dynamics, and Materials Conference, April 18-20, 1988, Williamsburg, Virginia, AIAA Paper No. 88-2216, pp. 62-74.

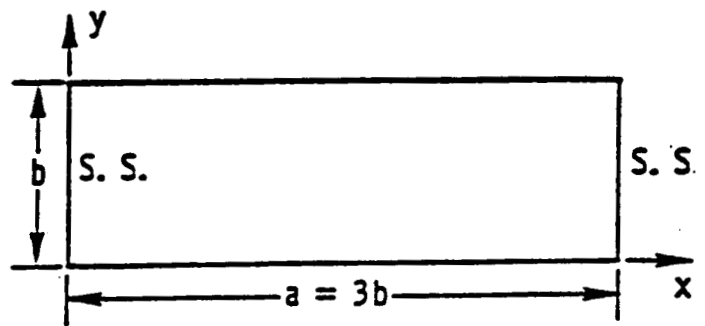
Table 1. Data for sample problem.

Property	Component	
	Skin	Stiffener
A_{11} , kips/in	1262	871
A_{12} , kips/in	379	261
A_{22} , kips/in	937	647
A_{66} , kips/in	381	263
D_{11} , in.-lb	742	244
D_{12} , in.-lb	223	733
D_{22} , in.-lb	551	181
D_{66} , in.-lb	224	738
Thickness, in.	0.084	0.058
Panel length, $a = 15.0$ in. Middle plate width = 5.0 in. Outer plate widths = 2.5 in. Stiffener height = 1.4 in. Load, $P = 15,792$ lb.		

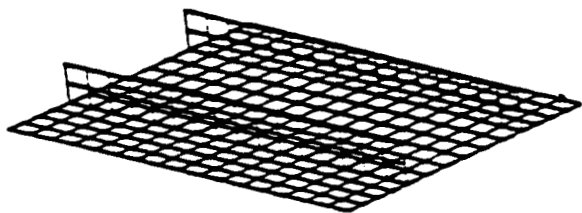
ORIGINAL PAGE IS
OF POOR QUALITY



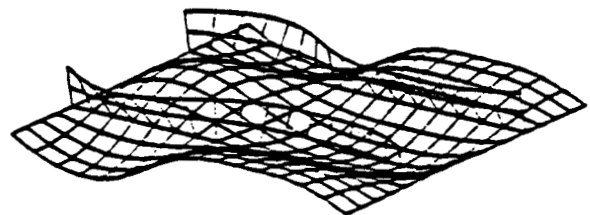
(a) Stiffened panel in compression.



(b) Coordinates for plate 2.

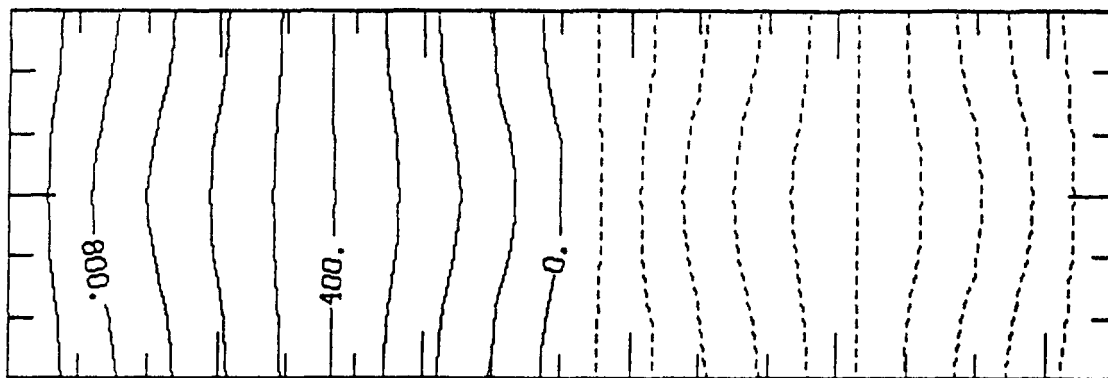


(c) Finite element grid.

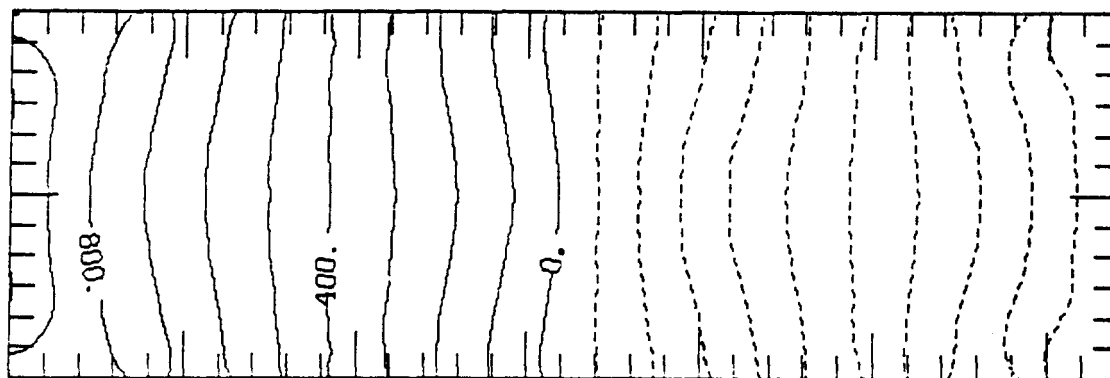


(d) Postbuckled deflection pattern.

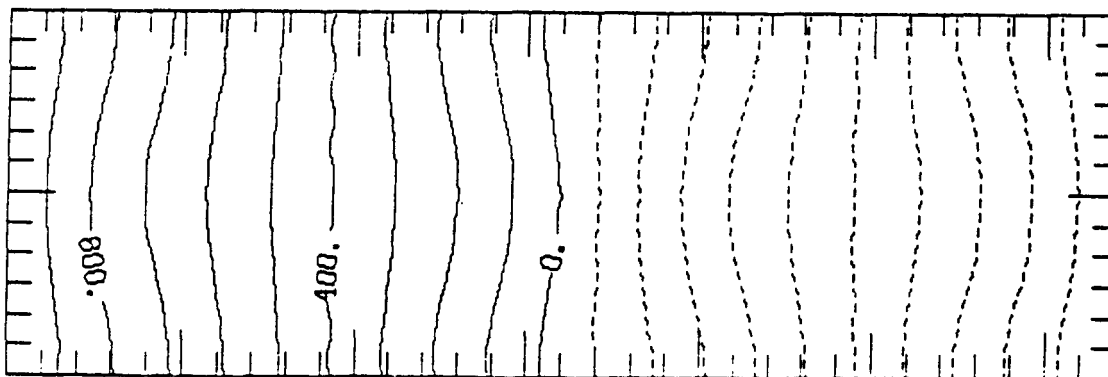
Fig. 1. Example of a plate component from a stiffened panel.



(a) Discrete FEM Solution

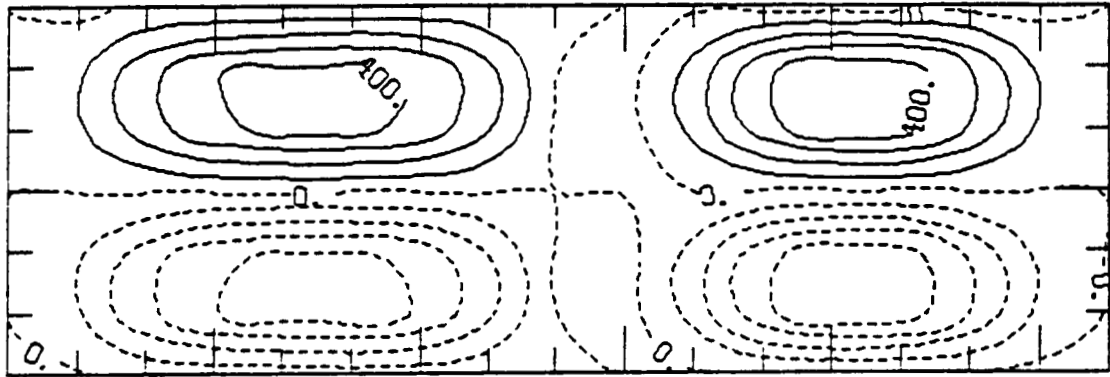


(b) Continuous Approximation (Ref. 4)

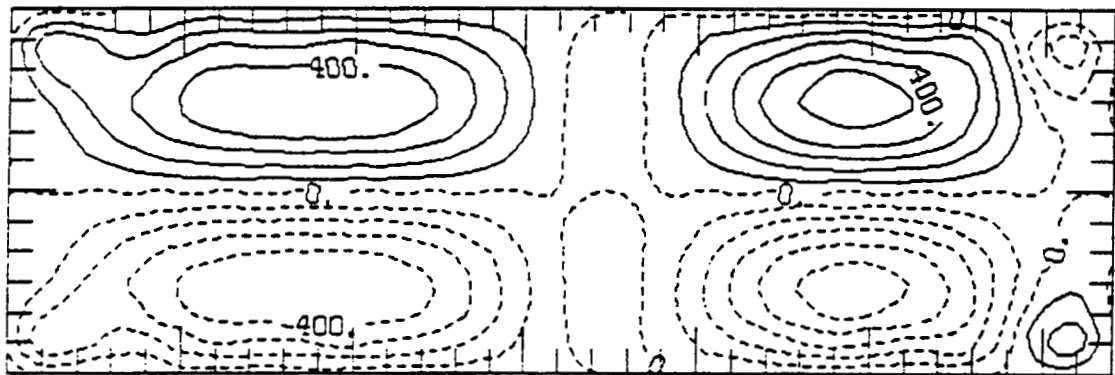


(c) Continuous Least Squares Approximation

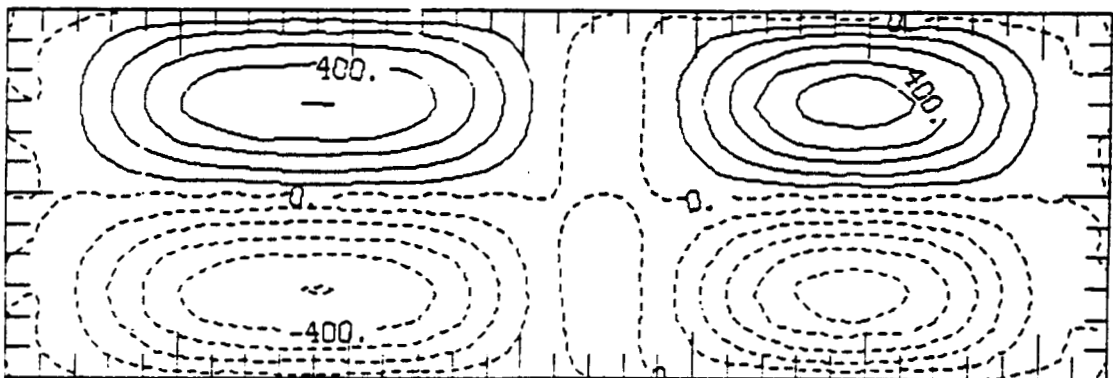
Fig. 2. In-plane Deflection, U , Contour Plots For Plate 2.



(a) Discrete FEM Solution

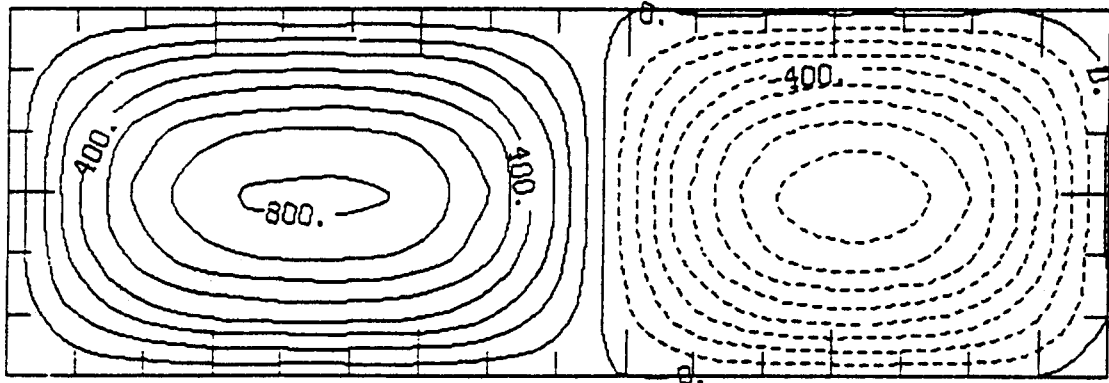


(b) Continuous Approximation (Ref. 4)

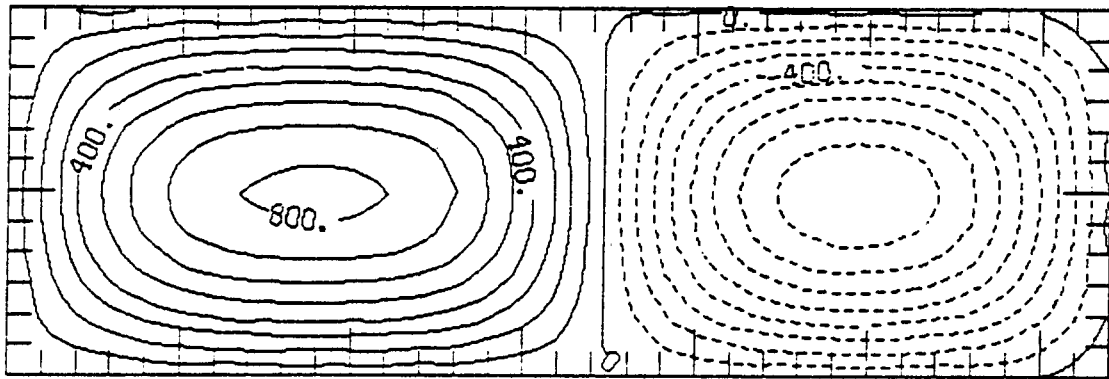


(c) Continuous Least Squares Approximation

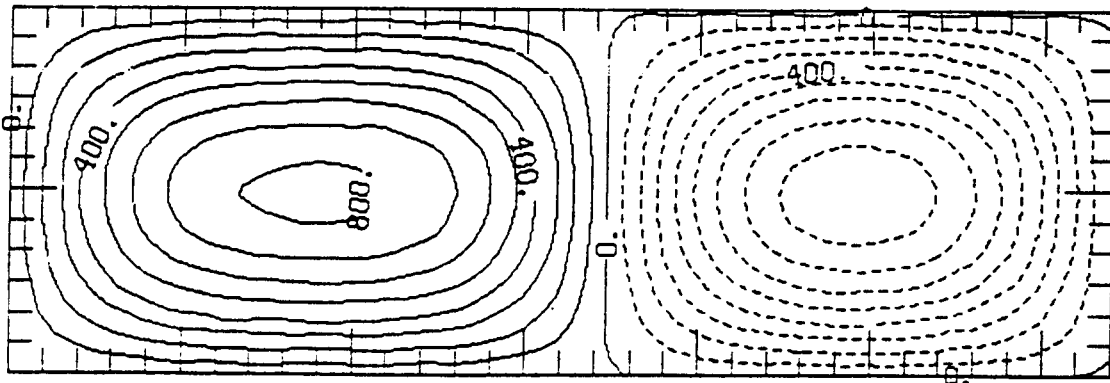
Fig. 3. In-plane Deflection, V , Contour Plots For Plate 2.



(a) Discrete FEM Solution

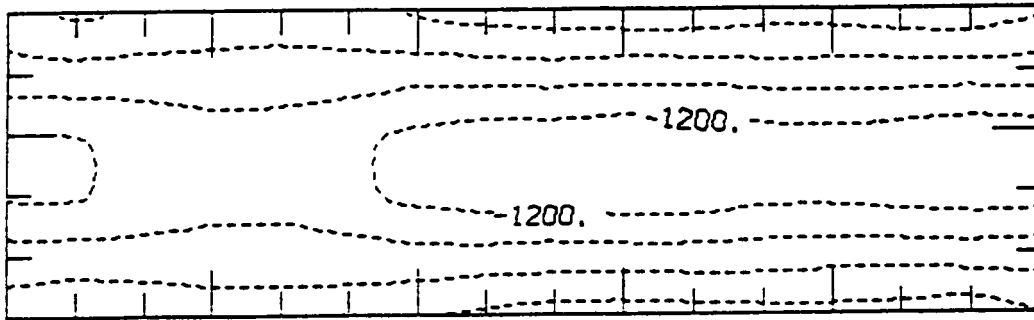


(b) Continuous Approximation (Ref. 4)

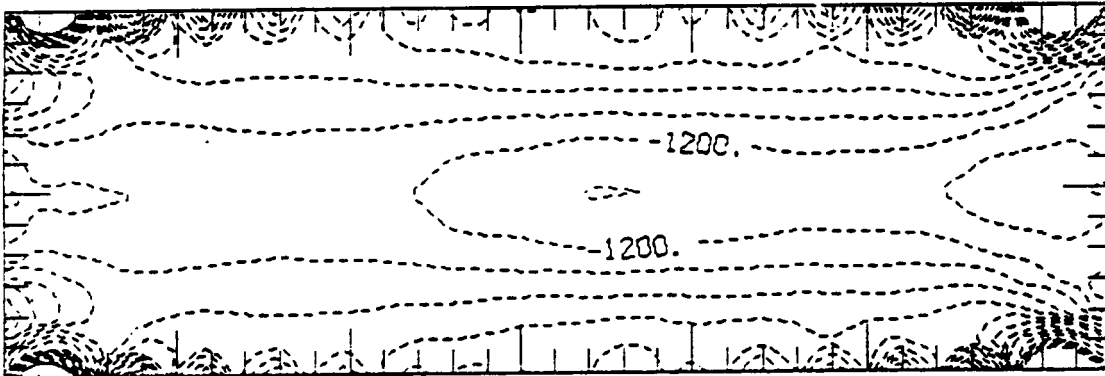


(c) Continuous Least Squares Approximation

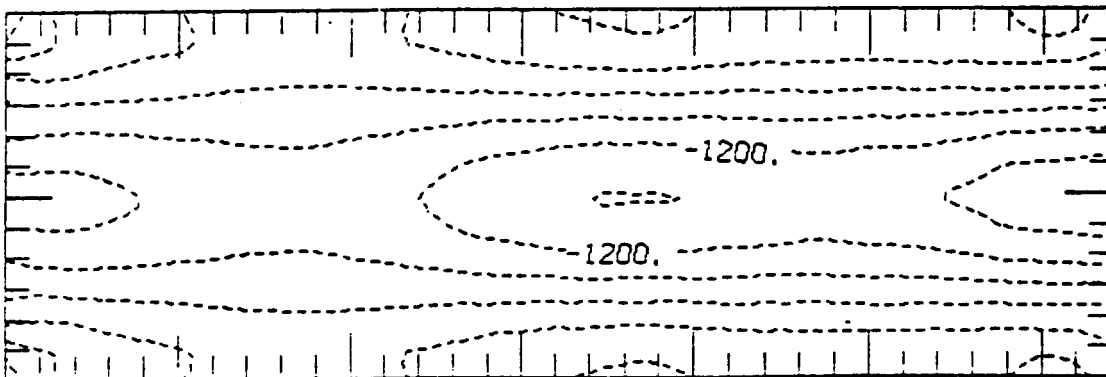
Fig. 4. Transverse Deflection, W , Contour Plots For Plate 2.



(a) Discrete FEM Solution



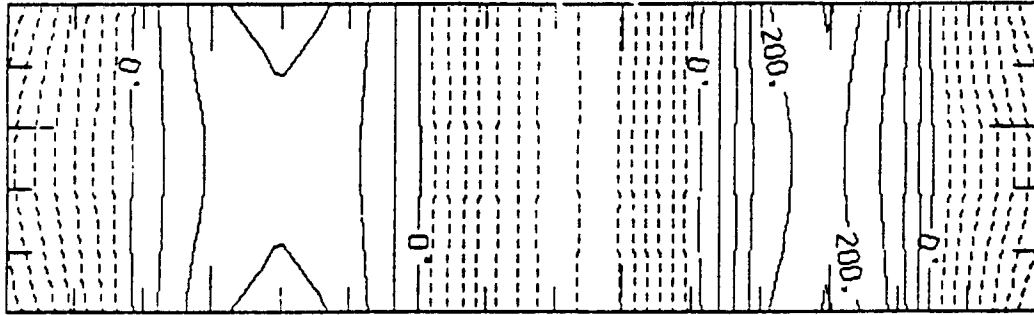
(b) Continuous Approximation (Ref. 4)



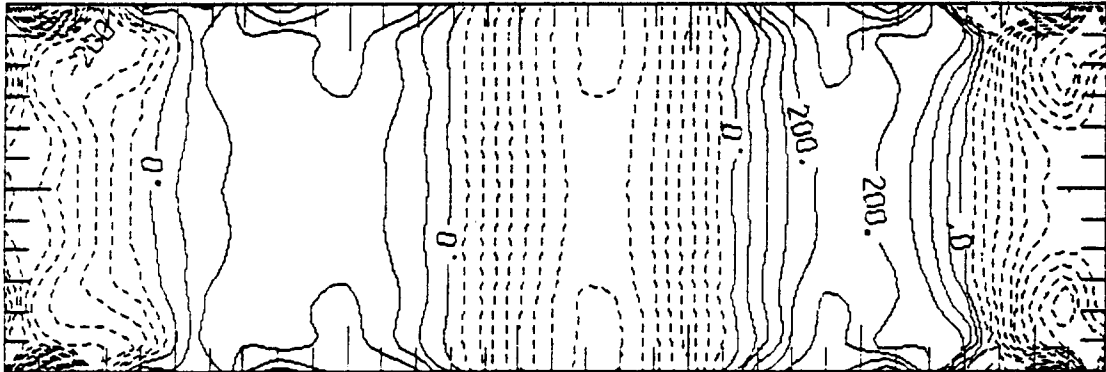
(c) Continuous Least Squares Approximation

Fig. 5. Stress Resultant, N_x , Contour Plots For Plate 2.

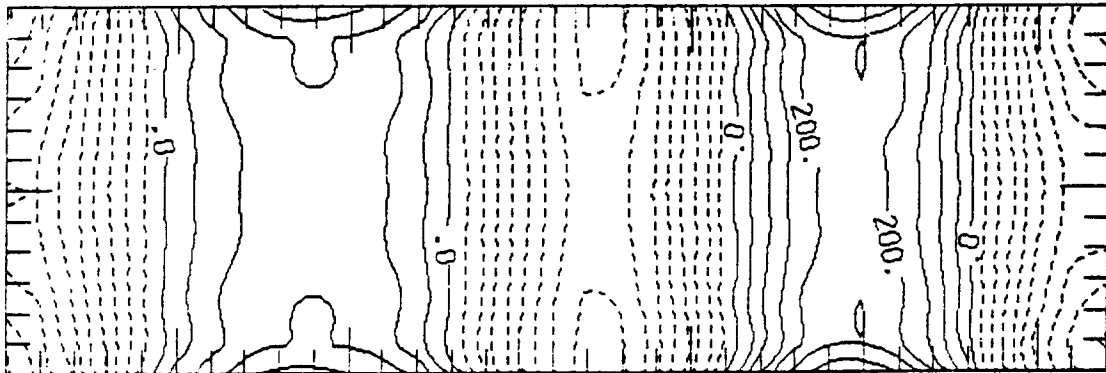
ORIGINAL PAGE IS
OF POOR QUALITY



(a) Discrete FEM Solution

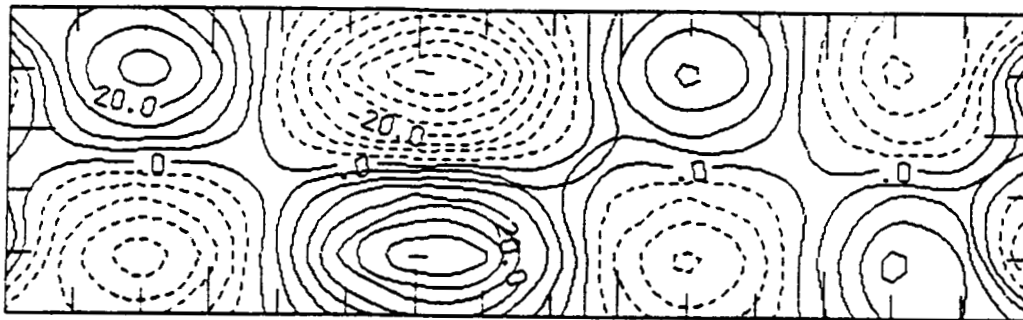


(b) Continuous Approximation (Ref. 4)

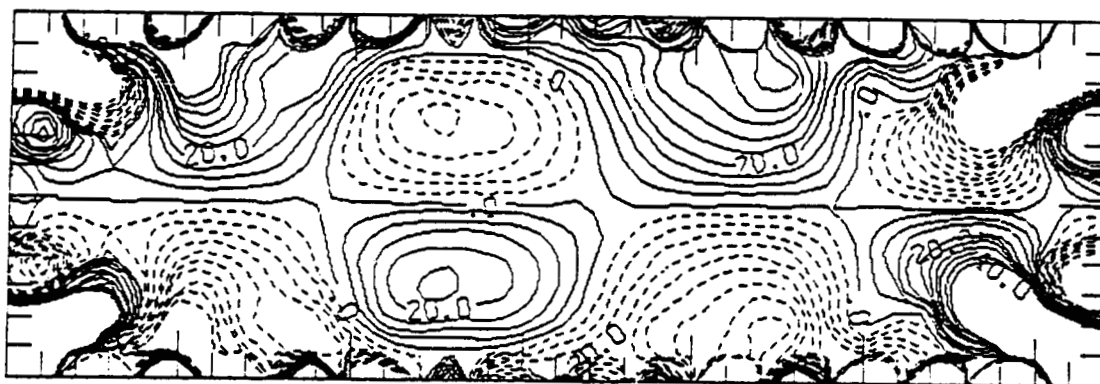


(c) Continuous Least Squares Approximation

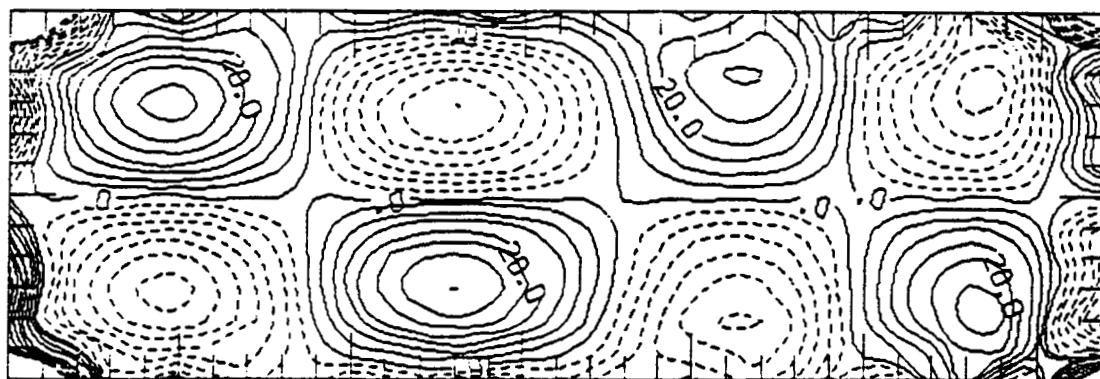
Fig. 6. Stress Resultant, N_y , Contour Plots for Plate 2.



(a) Discrete FEM Solution

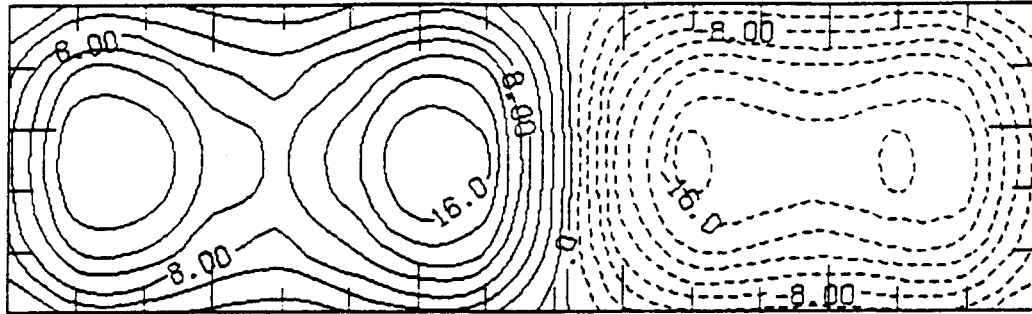


(b) Continuous Approximation (Ref. 4)

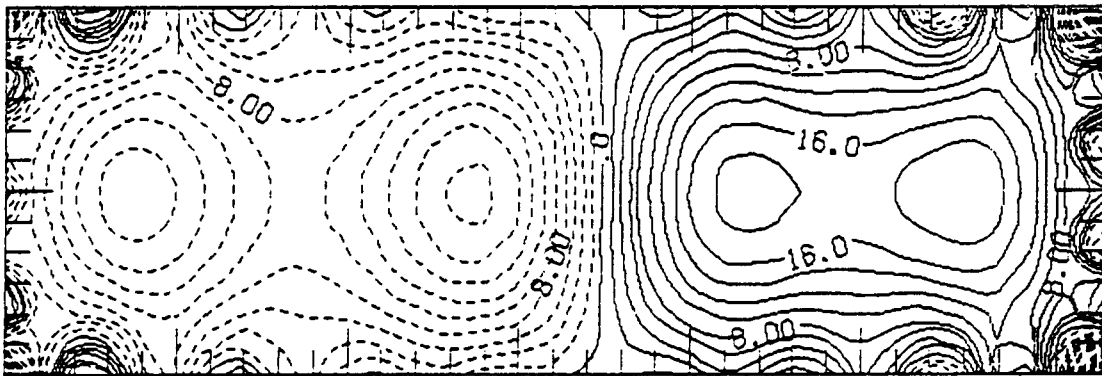


(c) Continuous Least Squares Approximation

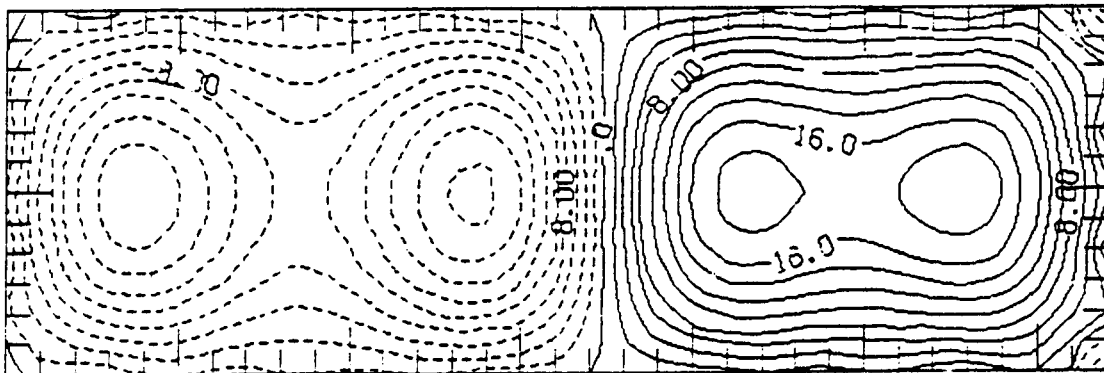
Fig. 7. Stress Resultant, N_{xy} , Contour Plots for Plate 2.



(a) Discrete FEM Solution



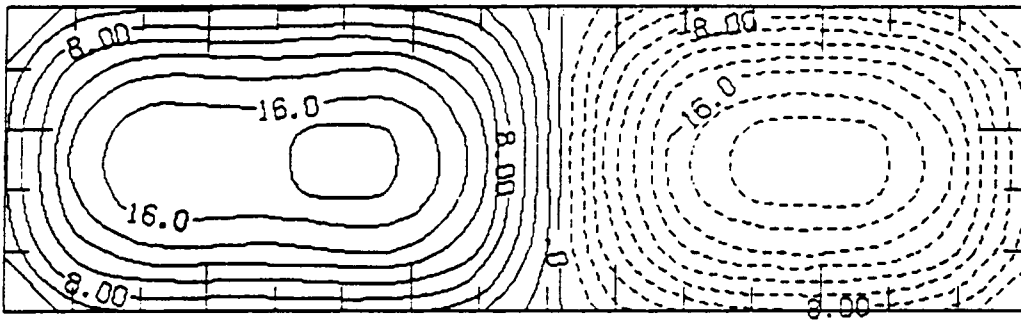
(b) Continuous Approximation (Ref. 4)



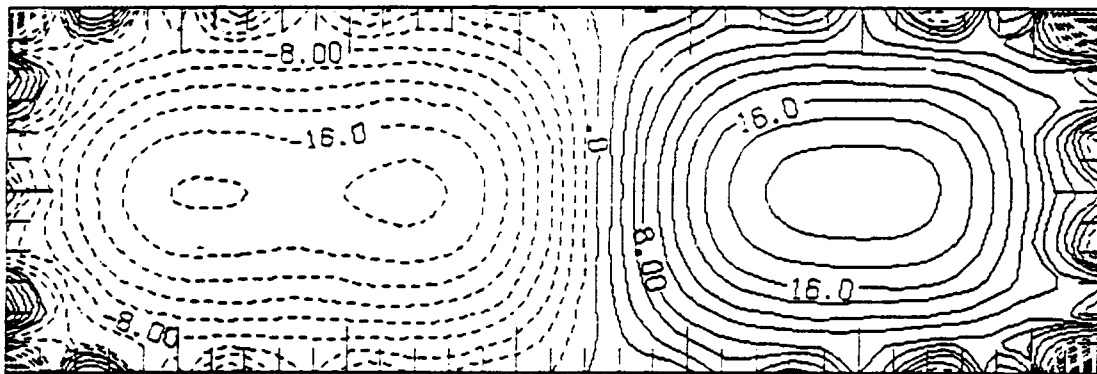
(c) Continuous Least Squares Approximation

Fig. 8. Moment Resultant, M_x , Contour Plots for Plate 2.

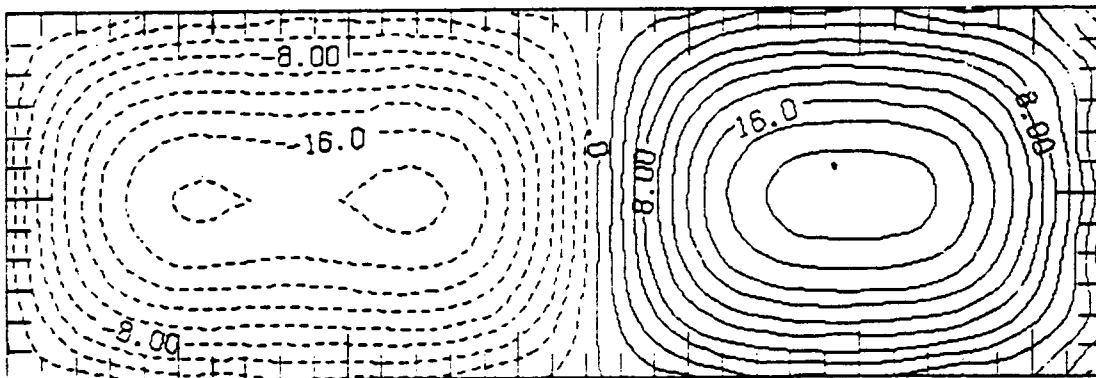
ORIGINAL PAGE IS
OF POOR QUALITY



(a) Discrete FEM Solution

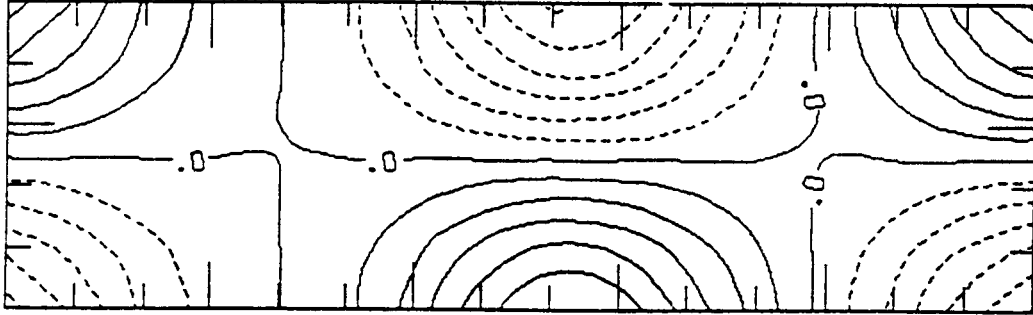


(b) Continuous Approximation (Ref. 4)

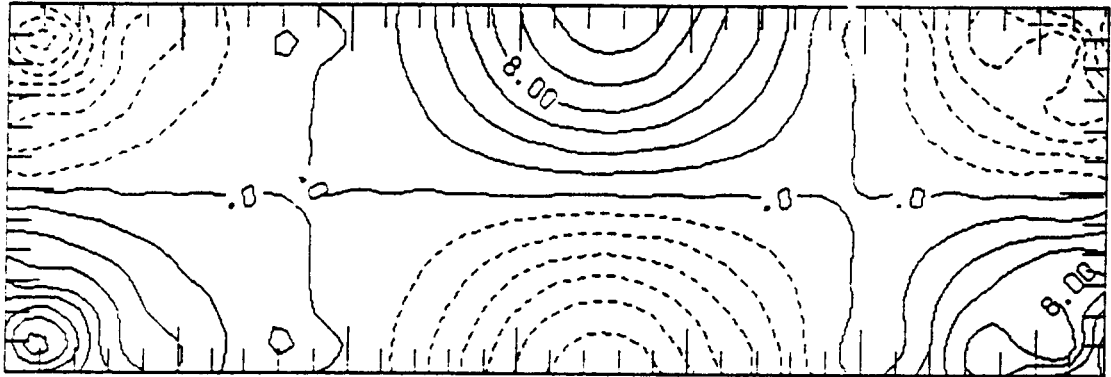


(c) Continuous Least Squares Approximation

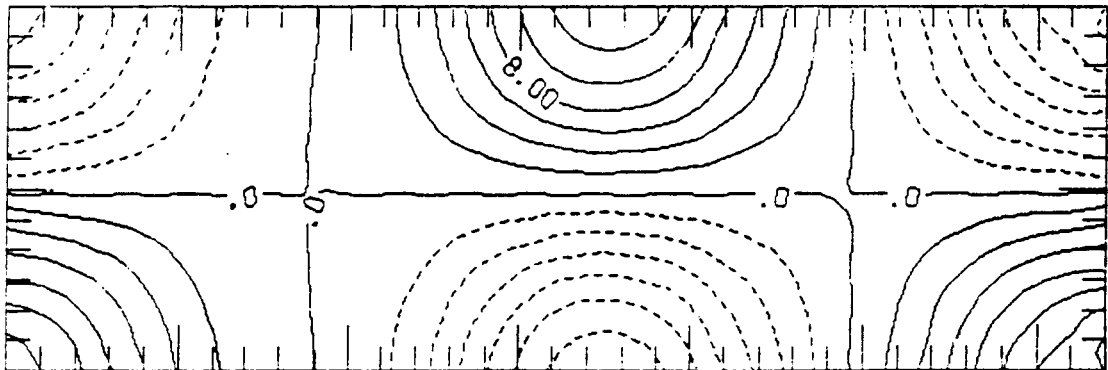
Fig. 9. Moment Resultant, M_y , Contour Plots for Plate 2.



(a) Discrete FEM Solution

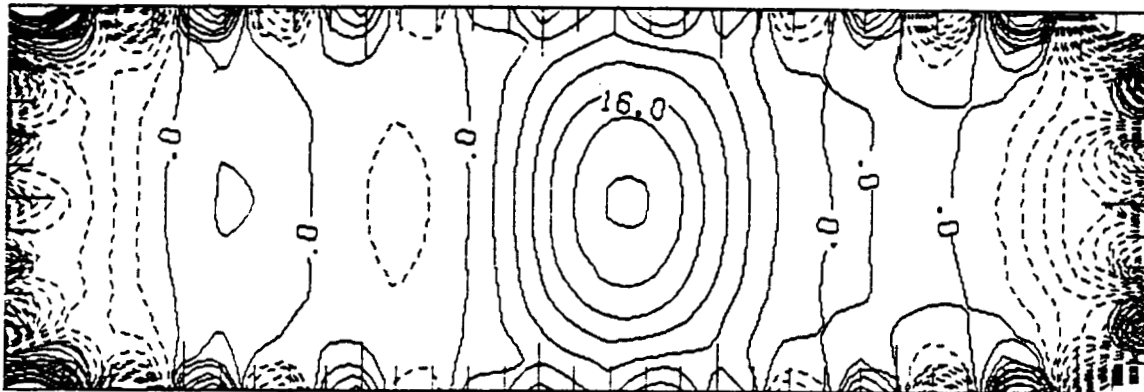


(b) Continuous Approximation (Ref. 4)

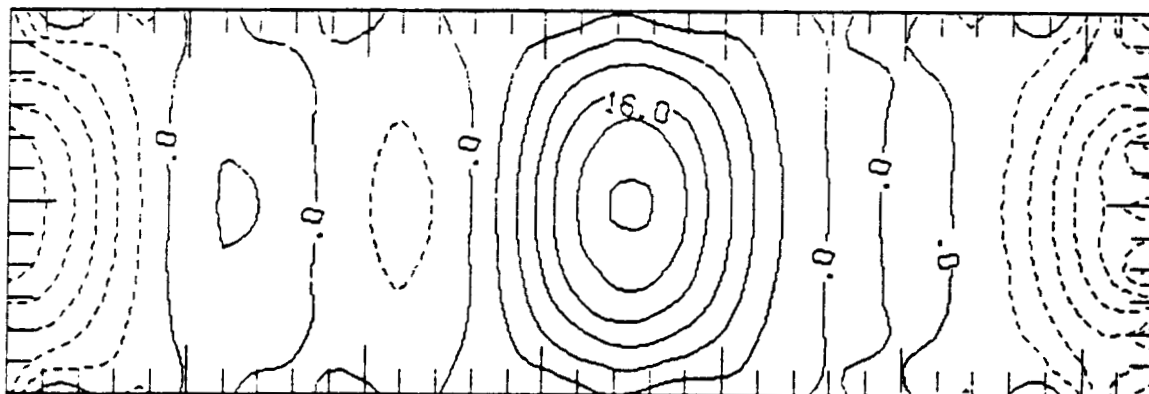


(c) Continuous Least Squares Approximation

Fig. 10. Moment Resultant, M_{xy} , Contour Plots for Plate 2.



(a) Continuous Approximation (Ref. 4)

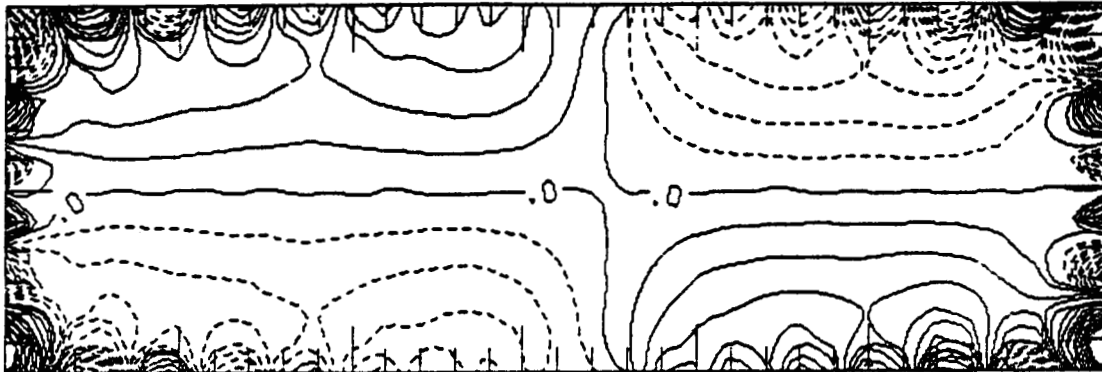


(b) Continuous Least Squares Approximation

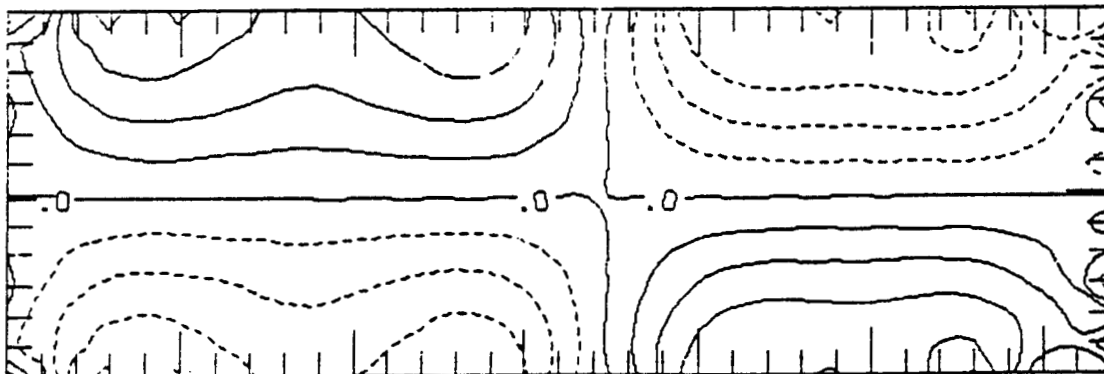
Fig. 11. Transverse Shear, Q_x , Contour Plots for Plate 2.

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY



(a) Continuous Approximation (Ref. 4)



(b) Continuous Least Squares Approximation

Fig. 12. Transverse Shear, Q_y , Contour Plots for Plate 2.

ORIGINAL PAGE IS
OF POOR QUALITY

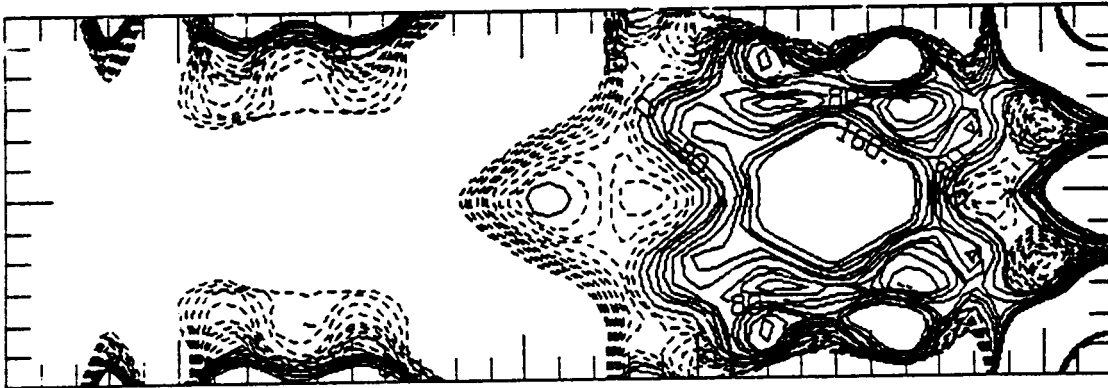


Fig. 13. Estimate Error in Transverse Equilibrium Equation.
(Error = $(L_{33} - N_3) / L_{33\max}$)



Report Documentation Page

1. Report No. NASA TM-100640	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle EAC: A Program for the Error Analysis of STAGS Results for Plates		5. Report Date January 1989	
		6. Performing Organization Code	
7. Author(s) Rajaram Sistla, Gaylen A. Thurston, and Nancy Jane C. Bains		8. Performing Organization Report No.	
		10. Work Unit No. 505-63-01-08	
9. Performing Organization Name and Address NASA Langley Research Center Hampton, VA 23665-5225		11. Contract or Grant No.	
		13. Type of Report and Period Covered Technical Memorandum	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546-0001		14. Sponsoring Agency Code	
15. Supplementary Notes Rajaram Sistla, AS&M, Hampton, VA. Gaylen A. Thurston and Nancy Jane C. Bains, Langley Research Center, Hampton, VA.			
16. Abstract A computer code is now available for estimating the error in results from the STAGS finite element code for a shell unit consisting of a rectangular orthotropic plate. This memorandum contains basic information about the computer code EAC (Error Analysis and Correction) and describes the connection between the input data for the STAGS "shell units" and the input data necessary to run the error analysis code. The STAGS code returns a set of nodal displacements and a discrete set of stress resultants; the EAC code returns a continuous solution for displacements and stress resultants. The continuous solution is defined by a set of generalized coordinates computed in EAC. The theory and the assumptions that determine the continuous solution are also outlined in this memorandum. An example of application of the code is presented and instructions on its usage on the Cyber and the VAX machines have been provided.			
17. Key Words (Suggested by Author(s)) error analysis finite elements plates		18. Distribution Statement Unclassified - Unlimited Subject Category 39	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of pages 72	22. Price A04